

**UNIVERSIDADE DE LISBOA**  
**Faculdade de Ciências**  
**Departamento de Informática**



**VISUALIZAÇÃO DE CONTEÚDOS GEOGRÁFICOS  
EM DISPOSITIVOS MÓVEIS**

**José Manuel Vieira Andrade**

**MESTRADO EM ENGENHARIA INFORMÁTICA**  
Especialização em Arquitectura, Sistemas e Redes de Computadores

2009



**UNIVERSIDADE DE LISBOA**  
**Faculdade de Ciências**  
**Departamento de Informática**



**VISUALIZAÇÃO DE CONTEÚDOS GEOGRÁFICOS  
EM DISPOSITIVOS MÓVEIS**

**José Manuel Vieira Andrade**

**PROJECTO**

Projecto orientado pela Prof. Dr.<sup>a</sup> Ana Paula Afonso

**MESTRADO EM ENGENHARIA INFORMÁTICA**  
Especialização em Arquitectura, Sistemas e Redes de Computadores

2009



## Agradecimentos

Aqui dedico o meu agradecimento a todos os que contribuíram e ajudaram no desenvolvimento deste projecto. A todos, os meus sinceros agradecimentos.

- Em primeiro lugar gostaria de agradecer à Prof. Dr.a Ana Paula Afonso pela forma como orientou o meu trabalho, pela disponibilidade que sempre dispôs e pela motivação que me transmitiu ao longo do meu percurso académico;
- Deixo também o meu agradecimento às professoras Beatriz Carmo e Ana Paula Cláudio pelo apoio e disponibilidade que me deram ao longo deste projecto;
- Os meus agradecimentos à FCT (Fundação para a Ciência e Tecnologia) através do projecto PTDC/EIA/69765/2006, por ter financiado o trabalho de investigação apresentado neste relatório;
- Sem me esquecer dos meus amigos, principalmente aqueles que me ajudaram desde o início do meu percurso académico e me deram todo o apoio durante o desenvolvimento deste projecto, aqui fica o meu agradecimento, ao meu amigo de muitos anos Leonardo, ao Hugo e ao Ivan;
- Agradeço também ao Javier pelo apoio e tempo despendido em relação a dúvidas pertinentes no decorrer deste projecto. A todos os meus colegas do XLDB aqui ficam os meus agradecimentos;
- Para a Raquel, os meus agradecimentos pelo apoio, simpatia, carinho e amizade que me proporcionou;
- Um especial agradecimento à minha família: principalmente às minhas irmãs pela amizade e apoio que me deram ao longo da minha vida e aos meus pais, que sempre se esforçaram para dar o melhor a todos os seus filhos;



*Aos melhores pais do mundo.*





## Resumo

As tecnologias normalizadas baseadas nos sistemas *Web Service*, têm vindo a facilitar o uso de dados geográficos na *Internet*, através do acesso, selecção e apresentação da informação. Apesar de permitir a transferência de grandes quantidades de informação, também é necessário facilitar a sua redução sobretudo em ambientes móveis, de acordo com as necessidades dos utilizadores, para que seja possível apresentar a informação de um modo mais eficiente. As limitações impostas pelos dispositivos móveis quando comparados com os computadores de secretária em termos de memória, autonomia, resolução, número de cores e velocidade de processamento, sugerem que é necessário determinar mecanismos que permitam ao utilizador obter e visualizar informação de modo eficaz e eficiente.

Este projecto descreve a experiência obtida e o trabalho realizado, na manipulação da informação geográfica, de modo a permitir a edição de referências visuais existentes num mapa. A manipulação da informação, refere-se à possibilidade de retirar ou adicionar camadas de informação. Estas camadas são acedidas em tempo real, de acordo com as preferências dos utilizadores, e sobre estas existe a possibilidade de seleccionar quais os atributos mais relevantes para a realização de uma dada tarefa. Deste modo, a análise e percepção dos dados torna-se menos confusa. Como proposta de solução, foi desenvolvido um protótipo. O desenvolvimento foi direccionado para a visualização da informação geográfica em dispositivos móveis. Os componentes essenciais da arquitectura seguem os modelos definidos pelo *OGC (Open Geospatial Consortium)*, *WMS (Web Map Service)*, *WFS (Web Feature Service)* e *SLD (Styled Layer Descriptor)*. Estes modelos normalizados permitem a edição e partilha de informação geográfica na *Web*, para além de permitirem seleccionar o modo como os dados são apresentados num mapa.

**Palavras-chave:** Visualização de Informação Geográfica, Dispositivos Móveis, *Web Services*, Especificação *WMS*, Especificação *SLD*.



## Abstract

The technologies based on standard Web Service systems, have facilitated the use of geographic data on the Internet through access, selection and presentation of geographic information. Although the transfer of large amounts of information is allowed, it is also necessary to simplify that information, specially in mobile environments, according to the needs of users, in order to present the information more efficiently. The restrictions imposed by mobile devices when compared to desktop computers in terms of memory, battery, resolution, number of colors and processing speed and power, suggest that it is necessary to establish mechanisms that allow the user to retrieve and view information effectively and efficiently.

This project describes the experience acquired and the work performed concerning the handling of geographic information, in order to enable the edition of existing visual references on a map. The manipulation of information refers to the possibility of removing or adding layers of information to a map. These layers are accessible in real time according to user preferences, and it is possible to select which attributes are most relevant to the performance of a particular task. Thus, the analysis and understanding of data becomes less confusing. In order to prove this concept, a prototype was developed. The development was directed to the visualization of geographic information on mobile devices. The essential components of the software architecture underlying the prototype developed are based on the formats defined by the OGC (Open Geospatial Consortium), WMS (Web Map Service), WFS (Web Feature Service) and SLD (Styled Layer Descriptor). These standard formats allow the sharing and edition of geographic information in the Web and provide several means for the presentation of geographical information on maps.

**Keywords:** Visualization of Geographic Information, Mobile devices, Web Services, specification WMS, specification SLD.



# Conteúdo

<b>Lista de Figuras</b>	<b>xiv</b>
-------------------------	------------

<b>Lista de Tabelas</b>	<b>xv</b>
-------------------------	-----------

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação . . . . .	1
1.2	Objectivos . . . . .	2
1.3	Contribuições . . . . .	3
1.4	Metodologia . . . . .	3
1.5	Estrutura do documento . . . . .	4
<b>2</b>	<b>Sistemas de Informação Geográfica</b>	<b>5</b>
2.1	Definição e Conceitos . . . . .	5
2.1.1	Informação . . . . .	5
2.1.2	Sistemas de Informação . . . . .	5
2.1.3	Sistemas de Informação Geográfica . . . . .	6
2.1.4	Representação de Dados Geográficos . . . . .	6
2.2	Visualização da Informação . . . . .	9
<b>3</b>	<b>Informação Geográfica na Web</b>	<b>15</b>
3.1	<i>Web Services</i> . . . . .	15
3.2	Especificações <i>Open Geospatial Consortium</i> . . . . .	16
3.2.1	Web Map Services . . . . .	16
3.2.2	<i>Web Feature Services</i> . . . . .	19
3.2.3	<i>Styled Layer Descriptior</i> . . . . .	23
3.3	Servidores de Mapas . . . . .	26
<b>4</b>	<b>Informação Geográfica em Dispositivos Móveis</b>	<b>29</b>
4.1	Enquadramento e trabalho relacionado . . . . .	29
4.2	Processo de Desenvolvimento . . . . .	35
4.3	Análise . . . . .	36

4.3.1	Modelo Dados . . . . .	36
4.3.2	Funcionalidades . . . . .	39
4.3.3	Desenho da Interface . . . . .	40
4.4	Desenho da arquitectura . . . . .	43
4.5	Implementação . . . . .	45
4.5.1	Ferramentas e Tecnologias Utilizadas . . . . .	46
4.5.2	Protótipo . . . . .	63
<b>5</b>	<b>Conclusões e trabalho futuro</b>	<b>71</b>
5.1	Conclusão . . . . .	71
5.2	Trabalho futuro . . . . .	72
5.3	Análise pessoal . . . . .	73
	<b>Abreviaturas</b>	<b>75</b>
	<b>Bibliografia</b>	<b>80</b>
	<b>Índice</b>	<b>81</b>

# Lista de Figuras

2.1	Geóide de representação da superfície terrestre. . . . .	10
2.2	Esferóide de representação da superfície terrestre. . . . .	10
2.3	Elipsóide de representação da superfície terrestre. . . . .	10
2.4	projecção plana. . . . .	13
2.5	projecção cilíndrica. . . . .	13
2.6	projecção cónica. . . . .	13
2.7	Ângulos de Latitude e Longitude. . . . .	13
2.8	Ângulos de Latitude segundo o modelo esférico da Terra. .	14
2.9	Ângulo de Latitude segundo o modelo elipsoidal da Terra.	14
3.1	Arquitectura Web Service . . . . .	16
3.2	Ficheiro <i>SLD</i> definindo duas regras de estilo. . . . .	26
4.1	Modelo de dados base do sistema MoViSys. . . . .	32
4.2	Arquitectura do sistema MoViSys. . . . .	33
4.3	Modelo evolutivo referente ao processo desenvolvimento do projecto. . . . .	36
4.4	Modelo de dados representando a camada pontos de inter- esse. . . . .	37
4.5	Modelo de dados representando a camada pontos de inter- esse e as restantes camadas referentes às <i>shapefiles</i> . . . . .	38
4.6	Diagrama Casos de Uso indicando operações de visualização.	41
4.7	Diagrama Actividades representando as interacções entre a interface, o utilizador e o sistema. . . . .	42
4.8	Modelo da Arquitectura do Sistema. . . . .	43
4.9	Fluxo de informação de um pedido <i>GetCapabilities</i> . . . . .	44
4.10	Fluxo de informação indicando pedido do mapa. . . . .	45
4.11	Fluxo de informação indicando operações de <i>zoom</i> . . . . .	46
4.12	Arquitectura do servidor <i>Geoserver</i> . . . . .	47
4.13	Página inicial do <i>GeoServer</i> . . . . .	48
4.14	Directorio de Dados do <i>GeoServer</i> . . . . .	49
4.15	Configuração do espaço de nomes. . . . .	50

4.16	Configuração do espaço físico para uma <i>shapefile</i> . . . . .	51
4.17	Configuração da localização de uma <i>shapefile</i> . . . . .	51
4.18	Configuração da ligação a uma base de dados <i>PostGis</i> . . .	52
4.19	Seleccção de uma camada para configuração. . . . .	53
4.20	Configuração da informação de uma camada de dados. . .	54
4.21	Configuração da informação de uma camada de dados. . .	54
4.22	Diagrama de classes da aplicação cliente . . . . .	57
4.23	Diagrama de sequência representando a interacção entre as componentes do sistema, no que se refere à geração e interpretação do ficheiro <i>SLD</i> . . . . .	60
4.24	Esquema hierárquico das regras do ficheiro <i>SLD</i> . . . . .	61
4.25	Exemplo de um ficheiro <i>SLD</i> . . . . .	62
4.26	Interface inicial. . . . .	63
4.27	Visualização do mapa. . . . .	63
4.28	Visualização do mapa num dado nível de aproximação. . .	65
4.29	Interface de selecção de opções. . . . .	65
4.30	Interface de selecção de camadas. . . . .	65
4.31	Interface obtida após a omissão da camada de estradas. . .	66
4.32	Visualização do mapa num dado nível de aproximação. . .	66
4.33	Interface de selecção de pontos de interesse. . . . .	67
4.34	Interface de selecção de pontos de interesse. . . . .	67
4.35	Interface de visualização de pontos de interesse no mapa. .	68
4.36	Visualização do mapa num dado nível de aproximação. . .	68
4.37	Interface de selecção das camadas a editar. . . . .	69
4.38	Interface de selecção das propriedades de uma camada. . . . .	69
4.39	Visualização do mapa num dado nível de aproximação. . .	69



# Lista de Tabelas

3.1	Parâmetros de um pedido <i>GetCapabilities</i> . . . . .	17
3.2	Parâmetros de um pedido <i>GetMap</i> . . . . .	19
3.3	Parâmetros de um pedido <i>GetFeatureInfo</i> . . . . .	20
3.4	Parâmetros comuns de um pedido <i>WFS</i> . . . . .	23



# Capítulo 1

## Introdução

### 1.1 Motivação

A visualização de informação geo-referenciada é uma área de investigação cujo objectivo essencial é melhorar a legibilidade e o discernimento dos dados de acordo com o contexto de utilização. Nos dispositivos móveis a visualização da informação é particularmente dificultada pelas capacidades limitadas dos mesmos em termos da velocidade de processamento, da autonomia, da resolução e do número de cores, dos mecanismos de interacção e da dimensão do ecrã quando comparadas com os computadores de secretária [42]. De forma a minimizar os efeitos destas restrições, é importante aplicar métodos de adequação dos dados a este novo meio de visualização.

A evolução dos dispositivos móveis, tem vindo a proporcionar novas oportunidades nos domínios onde a informação geográfica desempenha um papel importante. São exemplos a Cartografia, Turismo, Arqueologia, Planeamento Urbano, Sistemas de Navegação [47]. É necessário adoptar técnicas de visualização eficientes que permitam aos utilizadores aceder e compreender a informação necessária para a realização das suas tarefas. Para além de fornecer mecanismos de visualização eficientes, deve ser fornecido aos utilizadores ferramentas capazes de interagir com a informação geográfica, gerando conhecimento útil para suportar as suas decisões. A interacção fornecida por este tipo de aplicações para gerir os dados geográficos, são tipicamente operações de afastamento, aproximação e obtenção de informação sobre pontos de interesse sobre os mapas [44]. Contudo, a manipulação de referências visuais sobre um mapa, nomeadamente qual a informação geográfica visualizada a cada momento de acordo com as preferências do utilizador, ainda é um problema actual.

Um dos grandes problemas neste tipo de aplicações é a redução do número de elementos apresentados de modo a obter uma apresentação mais inteligível. A simplificação da informação apresentada nos mapas é importante, pois permite reduzir o esforço cognitivo do utilizador e simplificar a sua análise sobre a informação visualizada. Em [41] são utilizados mecanismos de filtragem, baseado em funções de grau de interesse, que permitem reduzir os elementos visualizados através de uma selecção dos elementos mais importantes. Para resolver os problemas de sobreposição dos símbolos sobre um mapa foi combinado uma técnicas de filtragem e operadores de generalização, capaz de agrupar elementos que se encontrem muito próximos.

O uso de mapas para representar a informação possui um papel importante no contexto dos dispositivos móveis, não só por proporcionar garantias de localização, como também por facilitar a interacção com os dados obtidos. Os mapas actualmente utilizados em muitos sistemas comerciais são imagens *bitmap* [19], como por exemplo, as utilizadas pelo sistema do *Google Maps*. A utilização de mapas neste formato coloca algumas limitações, nomeadamente na impossibilidade de editar, actualizar e modificar facilmente muitas referências visuais existentes. Assim, é essencial explorar técnicas de manipulação da informação para minimizar estas limitações.

## 1.2 Objectivos

Este projecto foi desenvolvido no contexto de um projecto de investigação, Visualização de Informação Geo-referenciada, que visa explorar técnicas de visualização de informação geo-referenciada sobre um mapa, quer para computadores de secretária quer para dispositivos móveis [8]. Os objectivos específicos deste projecto são:

- Compreender as principais vantagens e desvantagens da utilização de mapas em formato vectorial em relação ao uso de mapas em formato *bitmap*. Deste modo pretende-se provar conceptualmente as vantagens da manipulação de dados vectoriais no lado do servidor, permitindo aos utilizadores manipular as referências visuais obtidas nos mapas;
- Seleccionar de acordo com o estudo realizado, o formato vectorial mais adequado para a realização do projecto, relativamente à facili-

dade de editar as referências visuais obtidas (símbolos representando pontos de interesse e informação textual sobre os mesmos);

- Determinar os mecanismos adequados para obtenção e apresentação dos mapas dentro do contexto dos dispositivos móveis;
- Construção de um protótipo que permita aos utilizadores manipular e editar certas referências visuais presentes nos mapas, com base nos formatos estudados e os algoritmos de visualização concretizados em [46].

### 1.3 Contribuições

Da pesquisa e do trabalho desenvolvido relativos à visualização de conteúdos geográficos em dispositivos móveis foram concretizados os seguintes aspectos:

- Obtenção e apresentação da informação geográfica no contexto móvel, utilizando tecnologias normalizadas;
- Manipulação das camadas de informação geográfica por parte dos utilizadores, relativamente à informação visualizada;
- Opções de visualização no mapa referente à informação contida nas camadas de dados;
- Desenvolvimento de um protótipo como prova de conceito;

### 1.4 Metodologia

A metodologia adoptada baseou-se no modelo evolutivo de desenvolvimento. As fases foram executadas sequencialmente com refinamento das mesmas, com excepção do relatório final que foi sendo desenvolvido à medida que as fases anteriores foram sendo executadas. O trabalho foi decomposto do seguinte modo:

#### Trabalho de pesquisa e análise do problema

- Familiarização com o trabalho já realizado do sistema *MoViSys* e levantamento do trabalho relacionado através de uma pesquisa bibliográfica;

- Exploração dos formatos vectoriais e *bitmap* existentes;
- Realização de uma pesquisa de modo a determinar o melhor método para obtenção e apresentação da informação geográfica, no contexto dos dispositivos móveis;

#### **Desenvolvimento de um protótipo**

- Após a fase de análise foi desenvolvido um protótipo de acordo com os objectivos propostos, dando ênfase aos conceitos e tecnologias analisadas durante todo o projecto.

## **1.5 Estrutura do documento**

O resto do documento está organizado da seguinte forma:

- Capítulo 2 — Neste capítulo é apresentado um conjunto de conceitos, visando a compreensão do trabalho no contexto da informação geográfica;
- Capítulo 3 — Neste capítulo é apresentado um conjunto de conceitos referentes à manipulação da informação geográfica na *Web*, dando a conhecer conceitos e tecnologias aplicados no desenvolvimento desta dissertação;
- Capítulo 4 — Este capítulo descreve o mecanismo de visualização da informação geográfica em dispositivos móveis e o protótipo desenvolvido, dando a conhecer a sua arquitectura, o modelo de dados utilizado, as tecnologias utilizadas e os mecanismos empregues na visualização da informação;
- Capítulo 5 — Por último são apresentadas as conclusões do trabalho e o trabalho futuro.

## **Capítulo 2**

# **Sistemas de Informação Geográfica**

Neste capítulo são apresentados os conceitos essenciais sobre a informação geográfica e Sistemas de Informação Geográfica (SIG), que se enquadram no desenvolvimento deste projecto.

### **2.1 Definição e Conceitos**

#### **2.1.1 Informação**

Informação enquanto conceito, possui uma diversidade de significados, do uso quotidiano ao técnico. Genericamente, o conceito de informação está intimamente ligado às noções de restrição, comunicação, controle, dados, instrução, conhecimento, percepção e representação de conhecimento. A informação pode ser vista como o resultado do processamento, manipulação e organização de dados de tal forma que represente uma modificação (quantitativa ou qualitativa) no conhecimento de um sistema (pessoa, animal ou máquina) que a recebe. Etimologicamente a palavra informação significa, dar forma ou moldar a mente, como forma de educação, instrução ou treino [12].

#### **2.1.2 Sistemas de Informação**

Um Sistema de Informação pode ser designado como sendo uma abstracção da realidade, caracterizado por um conjunto de actividades que processam e utilizam a informação de um modo organizado. Estes sistemas podem ser automatizados, ou mesmo manuais, abrangendo pessoas e/ou máquinas. Existem vários tipos de Sistemas de Informação. De entre estes sistemas podemos encontrar: Sistemas de Suporte à Decisão, Sistemas de Gestão de Conhecimentos, Sistemas de Base de Dados, entre outros. Deste modo,

os sistemas de informação permitem manusear e processar a informação de forma organizada facilitando os processos cognitivos [24].

### 2.1.3 Sistemas de Informação Geográfica

Um Sistema de Informação Geográfica [25], é um sistema de informação que permite guardar, analisar e gerir dados que estão relacionados com localizações geográficas. Um Sistema de Informação Geográfica organiza a informação em diferentes camadas temáticas e armazena-as independentemente. Deste modo pode-se manipular a informação de uma forma rápida e simples, permitindo ao operador ou utilizador a possibilidade de relacionar a informação existente através da posição e topologia dos objectos. Outra definição designa estes sistemas como sendo um poderoso conjunto de ferramentas que permitem guardar, encontrar, transformar e mostrar dados espaciais para um determinado conjunto de objectivos [40].

As áreas de aplicação dos SIG, por serem muito versáteis, são muito vastas, desde a Cartografia, estudos de impacto Ambiental, Sistemas de Navegação, entre outros. A profunda revolução que provocaram as novas tecnologias afectou decisivamente a evolução da análise espacial. *Tobler* em 1970, desenvolveu a primeira lei da geografia, e foi o inventor das projecções sobre os mapas [26].

### 2.1.4 Representação de Dados Geográficos

Os dados representados referem-se a objectos do mundo real e podem ser armazenados usando dois tipos de representação abstracta: dados *bitmap* e dados vectoriais.

#### Dados *Bitmap*

Uma imagem gráfica referente a dados no formato *bitmap* [19] é constituída essencialmente por uma grelha rectangular de pontos de cor ou pixéis contidos numa estrutura de dados. Este formato de imagem é caracterizado pela largura, altura da imagem e pelo número de *bits* utilizado para representar cada pixel. As imagens deste tipo são armazenadas em variados formatos, tais como *JPEG*, *PNG*, *GIF*, que se distinguem pelo tipo de compressão utilizado e pelo número de bits suportado por pixel.



O número total de pixéis que compõem uma imagem definem a sua resolução, isto é, o seu nível de detalhe. O número de bits utilizado para representar a cor de cada pixel indica a gama de cores a ser utilizada. Estes dois factores determinam a qualidade da imagem, quanto maior for a qualidade da imagem maior é a quantidade de dados necessária para a representar. Um factor importante a referir é que o aumento arbitrário da resolução das imagens neste formato provoca a perda de qualidade. O motivo é que o efeito causado por este aumento faz com que os pixéis individuais que compõem a imagem passem a ser visíveis ao olho humano e consequentemente perde-se a noção de contornos que compõem a imagem.

**Formato JPEG:** O formato *JPEG* (*Joint Pictures Expert Group*) é um formato de imagem muito utilizado nas imagens fotográficas. Este formato utiliza um esquema de cores de 24 *bits*. Isto significa que aceita cerca de 16,8 milhões de cores. Apesar deste formato poder conter esta quantidade de cores, o olho humano não é capaz de as visualizar na totalidade. Deste modo, é possível retirar um conjunto de informações que representam a imagem e manter apenas as que são visíveis ao olho humano. Esse processo é conhecido como compressão e permite que imagens bastante realistas sejam criadas, ao mesmo tempo que diminui o tamanho dos dados. Este processo é favorável quando se pretende transferir ou disponibilizar as imagens na *Internet*. Neste caso o tipo de compressão denomina-se *lossy*, pois existe perda de qualidade no processo de compressão/descompressão.

**Formato PNG:** O formato *PNG* (*Portable Network Graphics*) é bastante semelhante ao formato *JPEG*, pois utiliza um esquema de 24 *bits* de cores. No entanto, em comparação com o *JPEG*, a compressão obtida normalmente é mais eficiente e não proporciona perda de qualidade, o que permite maior fidelidade em relação à imagem original. Além disso, as imagens em *PNG* não requerem muito espaço, podendo-se ter figuras de alta definição com um tamanho pequeno. Ainda assim, o *JPEG* permite obter imagens de menor tamanho em alguns casos.

**Formato GIF:** O formato *GIF* (*Graphics Interchange Format*) é um formato de imagens que utiliza um esquema de 8 *bits* para representar as suas imagens, isto é, pode trabalhar com um gama de 256 cores. Devido a esta característica não é aconselhável no uso de imagens que necessitam de uma aproximação realística de grande definição. Este formato é bastante

utilizado na representação de ícones ou em imagens que não necessitam de uma grande resolução.

## Dados Vectoriais

Uma imagem gráfica referente a dados no formato vectorial [30] é baseada em primitivas geométricas, isto é, equações matemáticas que descrevem elementos bidimensionais (2D) tais como: texto, pontos, linhas, curvas e polígonos ou objectos tridimensionais (3D) tais como, sólidos ou outros volumes. Estes objectos são baseados em fórmulas matemáticas e são usados para representar uma imagem. Ao contrário das imagens *bitmap*, as imagens no formato vectorial não perdem qualidade quando sofrem mudanças de escala. A resolução da imagem pode ser aumentada para qualquer tamanho e mesmo assim manter a sua nitidez. O motivo baseia-se no facto de um reajuste no tamanho da imagem implicar somente uma adaptação das equações matemáticas, em vez da própria imagem. Outro factor importante está relacionado com o tamanho dos ficheiros neste formato, alterações na escala das imagens não tem grande impacto na alteração do tamanho das imagens.

**SVG:** O formato *SVG (Scalable Vector Graphics)* [20] é baseado na linguagem *XML*<sup>1</sup> usada para descrever imagens gráficas em duas dimensões. O aparecimento desta linguagem veio proporcionar aos serviços que usam mapas para representar a informação, uma maior qualidade na apresentação de imagens gráficas, como também facilitar a interacção como os mesmos. O *SVG* é uma recomendação do consórcio *W3C*<sup>2</sup> desde 2001 e por esse motivo tem ganho grande popularidade nos últimos anos.

Empresas como a *IBM*, a *Apple*, a *Adobe*, a *Kodak*, deram o seu contributo no desenvolvimento desta tecnologia. Contudo, muitas companhias atrasaram o suporte à linguagem *SVG* e outras não forneceram o seu apoio com receio de concorrência com alguns dos seus produtos comerciais [45]. O *SVG* é baseado na linguagem *XML* e por isso pode beneficiar de outras tecnologias relacionadas com *XML* [35], como por exemplo: *HTML*<sup>3</sup> e

---

<sup>1</sup>*XML - eXtensible Markup Language*. É uma recomendação da *W3C*, que descreve uma linguagem utilizada para representar classes de objectos, capaz de ser interpretado por computadores.

<sup>2</sup>*W3C - Desígnio de - World Wide Web Consortium*. É um consórcio de empresas de tecnologia que desenvolve padrões com o objectivo de criar e interpretar conteúdos na Web.

<sup>3</sup>*HTML - HyperText Markup Language*, é uma linguagem de definição de documentos utilizada para produzir páginas Web.

*XHTML*<sup>4</sup>.

A versão 1.0 do *SVG* foi recomendada em Setembro de 2001. Em Janeiro de 2003 foi recomendada a versão 1.1 com o objectivo de minimizar algumas restrições impostas no uso dos dispositivos móveis. A versão *SVG Tiny* foi recomendada para facilitar o uso em telefones móveis e o *SVG basic* para dispositivos móveis com mais capacidades, como os *PDA* e *Smartphones*. A versão 1.2 continua em desenvolvimento, com melhorias em relação às versões anteriores, como por exemplo, maior suporte para acções de outro tipo de média (áudio e vídeo).

**Shapefile:** *Shapefile* [22] é um formato de dados vectorial usado nos Sistemas de Informação Geográfica, para representar geometrias e seus respectivos atributos. Foi introduzido no início dos anos 90 e desenvolvido pela *ESRI*<sup>5</sup>. Os dados suportados por este formato são essencialmente primitivas geométricas tais como: pontos, linhas e polígonos. Os atributos associados a estas primitivas descrevem o que representam, por exemplo, uma linha pode representar uma estrada ou um rio. O *shapefile* é constituído por um conjunto de três ficheiros, que possuem o mesmo nome e são diferenciados pelas suas extensões. Os dados geométricos são armazenados em ficheiros com extensão *.shp*, denominado ficheiro principal e os atributos são armazenados num ficheiro com extensão *.dbf*. Esta extensão é proprietária e pertence ao *Xbase* [33]. O terceiro ficheiro com extensão *.shx* é um índice espacial e contém o endereço e o tamanho de cada registo do ficheiro principal (*.shp*).

## 2.2 Visualização da Informação

O uso de mapas como modo de visualização da informação, veio proporcionar aos utilizadores, uma maior facilidade de compreensão e análise da informação geográfica. Os mapas podem ser usados para os mais variados propósitos, como navegação, visualização de pontos de interesse, demarcação territorial, entre outros. A utilização do mapa tem como objectivo fornecer informação geográfica de um modo mais perceptível ao utilizador.

<sup>4</sup>XHTML - eXtensible Hypertext Markup Language, é uma reformulação da linguagem HTML baseada em XML. Combina as *tags* de marcação HTML com regras XML.

<sup>5</sup>ESRI (*Environmental Systems Research Institute*), é uma companhia de desenvolvimento de software e que fornece serviços na área dos sistemas de informação geográfico. Foi fundada em 1969.

## Mapas

Os mapas são uma representação reduzida e simplificada da superfície da terra ou parte dela, num plano. Permite representar posições e localizações do globo terrestre. Para representar a superfície terrestre num determinado plano é necessário aplicar um conjunto de procedimentos, que visa relacionar os pontos da superfície terrestre a pontos correspondentes no plano (mapa).

Sabe-se que o globo terrestre não é uma esfera mas sim um Geóide [2]. Como a irregularidade geométrica da Geóide torna complexa a sua expressão analítica, considera-se outra superfície que seja de expressão analítica mais simples e significativamente próxima desta, o Elipsóide. As Figuras 2.1, 2.2 e 2.3 representam os modelos de aproximação da superfície terrestre.

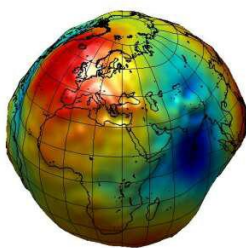


Figura 2.1: Geóide de representação da superfície terrestre.

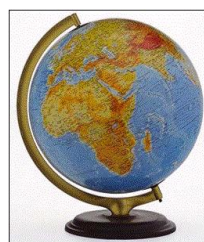


Figura 2.2: Esferóide de representação da superfície terrestre.



Figura 2.3: Elipsóide de representação da superfície terrestre.

Com o propósito de fundamentar a questão prática em torno da visualização da informação geográfica, apresenta-se de seguida uma revisão dos aspectos conceptuais mais importantes, Sistemas de Coordenadas, *Datum*, elipsóides de referência e projecção cartográfica, finalizando-se esta etapa com definições formais de latitude e longitude geodésicas.

## Sistema de Coordenadas

A superfície terrestre pode ser descrita geometricamente a partir de levantamentos geodésicos [3] ou topográficos [13] tendo como base sistemas de coordenadas distintos. Estes sistemas servem como referência para o posicionamento de pontos sobre uma superfície de referência, um elipsóide, uma esfera ou um plano. Os sistemas de coordenadas geodésicas utilizam um elipsóide ou uma esferóide como superfície de referência, dependendo da propriedade que se quer manter, pois é difícil garantir que não existam deformações quando se realiza uma transformação de um sistema de coordenadas geográficas, latitude e longitude, para coordenadas planas,  $x$  e  $y$ . Por exemplo, pode-se querer garantir os ângulos reais entre os dados apresentados, ou as distâncias. Por outro lado, o sistema de coordenadas plano utiliza uma superfície de referência plana, por exemplo, um cone ou um cilindro.

## Sistemas de Coordenadas Geodésico

Este sistemas de coordenadas permite determinar qualquer localização referente à superfície terrestre, baseando-se nos conceitos de latitude e longitude. A informação sobre como é definido um sistema de coordenadas geográfico encontra-se no que se denomina de *Datum*.

### *Datum*

O *Datum* contém informação do elipsóide de referência, que define os semi-eixos do sistema de coordenadas, referentes à latitude e longitude. O elipsóide de referência é uma superfície matematicamente definida que se aproxima da Geóide, a verdadeira figura da Terra ou qualquer outro corpo planetário. Para além da definição dos eixos que representam o sistema de coordenadas, existe informação sobre a localização inicial do elipsóide, denominado de ponto central. O *Datum* possui ainda informação sobre as unidades em que estão representados os dados, as unidades podem ser metros ou graus, dependendo do sistema de coordenadas utilizado.

## Sistema de Coordenadas Cartesiano

O sistema de coordenadas cartesiano, também conhecido por sistema de coordenadas planas, baseia-se em dois eixos perpendiculares, usualmente os eixos horizontal e vertical, cuja intersecção é denominada origem. Esta

origem é estabelecida como base para a localização de qualquer ponto do plano. Nos Sistemas de Informação Geográfica [23] as coordenadas planas normalmente representam uma projecção cartográfica, portanto são relacionadas matematicamente às coordenadas geográficas, de maneira que umas podem ser convertidas nas outras. Um sistema de referência cartesiano para além do *Datum* possui informação sobre a projecção. A projecção indica um conjunto de informação necessária para a transformação de dados geográficos num referencial cartesiano. A junção entre um *Datum* e a projecção designa-se por um sistema de projecção cartográfica.

## Projecção

Uma projecção é um procedimento utilizado para transformar coordenadas geodésicas (latitude e longitude) em coordenadas planas de modo a manter as relações espaciais. Todos os mapas são representações aproximadas da superfície terrestre. Isto ocorre porque não se pode passar de uma superfície curva para uma superfície plana sem que haja deformações. A elaboração de um mapa consiste em um método segundo o qual se faz corresponder a cada ponto da Terra, em coordenadas geográficas, um ponto no mapa, em coordenadas planas. Para se obter essa correspondência utilizam-se os sistemas de projecções cartográficas. Existem diferentes projecções cartográficas, uma vez que há uma variedade de modos de projectar sobre um plano os objectos geográficos que caracterizam a superfície terrestre. Uma projecção contém a seguinte informação:

**Longitude de origem:** Indica um meridiano de referência escolhido para posicionar o eixo y do sistema de coordenadas planas, que corresponde ao zero.

**Latitude de origem:** Contém também um paralelo de referência escolhido para posicionar o eixo x do sistema de coordenadas planas.

**Escala:** Escala é a relação matemática entre as dimensões dos elementos representados num mapa e a sua medida real na superfície terrestre. Classificam-se as projecções em: plana ou azimutal, cilíndricas e cónicas, segundo se represente a superfície curva da Terra sobre um plano, um cilindro ou um cone tangente à Terra. As Figuras 2.4, 2.5, 2.6, descrevem exemplos dessas projecções [11].

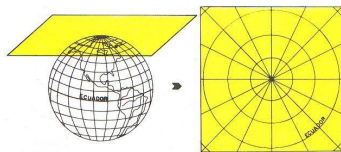


Figura 2.4: Projecção plana.

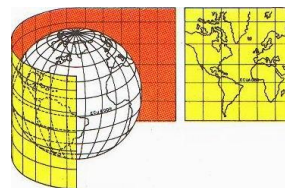


Figura 2.5: Projecção cilíndrica.

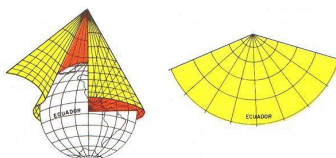


Figura 2.6: Projecção cónica.

## Coordenadas Geográficas

A coordenada geográfica de um objecto ou local da superfície terrestre é representada por dois valores, latitude e longitude. A Figura 2.7 exemplifica como são medidos esses valores.

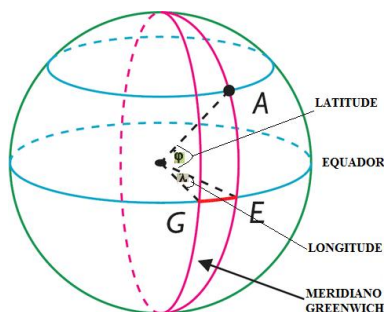


Figura 2.7: Ângulos de Latitude e Longitude.

**Latitude:** A Latitude define o ângulo que um objecto ou local da superfície terrestre realiza com o plano do equador. Este ângulo pode ser expresso em graus, minutos e segundos. Sistemas como o *Google Maps* podem expressar este valor em graus decimais.

Num modelo esférico da Terra, a latitude de um lugar é o ângulo que o raio que passa por esse lugar faz com o plano do equador. Uma vez que o raio da esfera é constante, este valor é também igual ao ângulo realizado entre o equador e o meridiano do lugar onde se encontra o objecto. Um exemplo de uma medida de um ângulo de latitude pode ser observado na Figura 2.8.

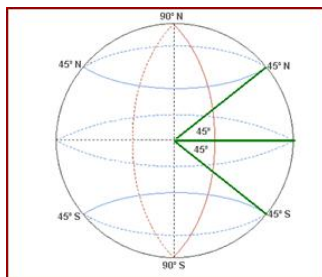


Figura 2.8: Ângulos de Latitude segundo o modelo esférico da Terra.

Num modelo elipsoidal da Terra, a latitude de um lugar é o ângulo que a recta perpendicular(normal) ao elipsóide nesse lugar faz com o plano do equador. Ao contrário do que acontece com o modelo esférico da Terra, as rectas normais ao elipsóide nos vários lugares não são todas concorrentes no centro da Terra. Por outro lado, e devido ao facto de os meridianos [15] não serem circunferências, mas sim elipses, a latitude não pode ser confundida, como na esfera, com a medida angular do arco de meridiano entre o equador e o lugar. As latitudes dos lugares representados nos mapas são latitudes geodésicas, como mostra o exemplo da Figura 2.9.

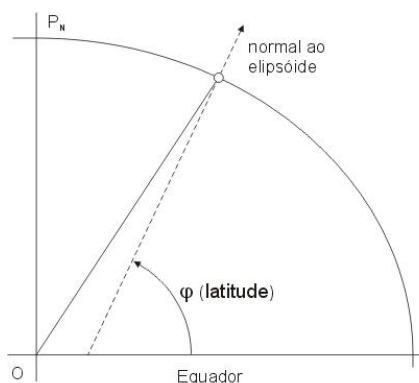


Figura 2.9: Ângulo de Latitude segundo o modelo elipsoidal da Terra.

**Longitude:** A Longitude define o ângulo que um objecto ou local da superfície terrestre realiza com um meridiano principal. Este valor varia entre 0 graus e 180 graus para Este ou para Oeste. Diferentemente da latitude, que tem a linha do Equador como um marco inicial natural, não há uma posição inicial natural para marcar a longitude. Portanto, um meridiano principal de referência tinha que ser escolhido. Foi convencionado o Meridiano de *Greenwich* como referência.



## Capítulo 3

# Informação Geográfica na Web

Neste capítulo apresentam-se as tecnologias e componentes que envolvem a manipulação da informação geográfica na *Web*, com o objectivo de fornecer um enquadramento de utilização no projecto desenvolvido. A primeira parte é iniciada com uma descrição dos *Web Services*. Na segunda parte apresentam-se as especificações *WMS (Web Map Service)* e *WFS (Web Feature Service)*, descrevendo a linguagem *SLD (Styled Layer Descriptor)*, finalizando o capítulo com a descrição de alguns servidores de mapas.

### 3.1 *Web Services*

Os *Web Services* são componentes que permitem às aplicações a troca de dados em formato *XML*. Estas componentes permitem com que os recursos de uma determinada aplicação estejam disponíveis sobre uma rede como a *Internet*, de uma forma normalizada. A grande vantagem do uso dos *Web Services* é a interoperabilidade entre diferentes máquinas ou aplicações.

Através das tecnologias *Web Services*, uma aplicação pode invocar outra para realizar tarefas simples ou complexas, mesmo que estas se encontrem escritas em diferentes linguagens ou se encontrem em diferentes sistemas.

Os *Web Services* são identificados por um *URI (Unique Resource Identifier)*, descritos e definidos usando a linguagem *XML*. O acesso aos *Web services* definem uma arquitectura cliente servidor, utilizando o protocolo *SOAP (Simple Object Access Protocol)*, que utiliza a linguagem *XML* e baseia-se na invocação remota de um método. Os parâmetros da invocação são codificados através deste protocolo e os dados são enviados pela rede utilizando o protocolo *HTTP (Hypertext Transfer Protocol)*. O pedido é processado no servidor e a resposta é enviada para o cliente.

Apresenta-se na Figura 3.1 a descrição da arquitectura *Web Service*: O *Server Provider* é o responsável por publicar os serviços existentes, utilizando a linguagem *WSDL* (*Web Services Description Language*). O responsável pelo registo e localização dos serviços na *Web* é o *Service Broker* através do protocolo *UDDI* (*Universal Description Discovery and Integration*), enquanto que o *Service Requestor* representa o utilizador dos serviços. O cliente e o servidor comunicam utilizando o protocolo *SOAP*.

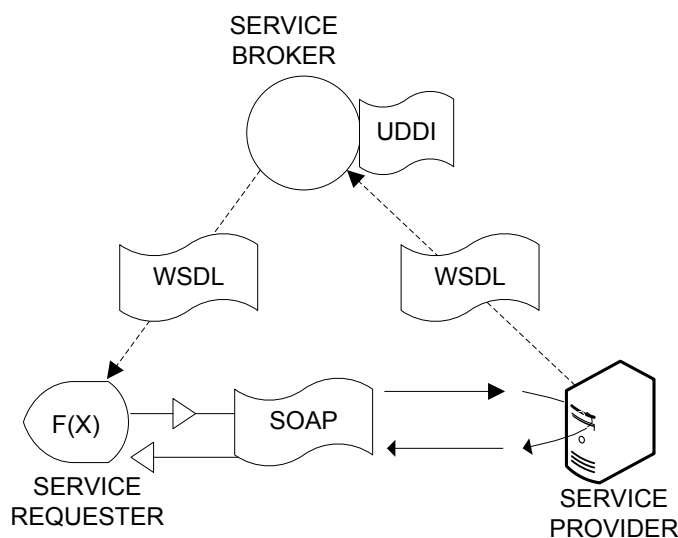


Figura 3.1: Arquitectura Web Service

## 3.2 Especificações *Open Geospatial Consortium*

### 3.2.1 Web Map Services

*Web Map Services* [32] é uma especificação definida pelo *OGC* (*Open Geospatial Consortium*), que tem como objectivo fornecer uma interface normalizada para o acesso a mapas. A interface fornecida é uma simples interface *HTTP*, que permite a aplicações clientes obter mapas em forma de imagem. Os dados obtidos provêm de informação geográfica existente. Estes mapas são fornecidos por um servidor designado de *Map Server*, que usa informação proveniente de uma base de dados *SIG*. Os mapas obtidos são uma representação visual dos dados e não os dados propriamente ditos. Estes mapas podem ser representados no formato *bitmap* ou no formato vectorial.

Esta especificação define três operações. A primeira descreve um serviço de informação sobre os dados existentes no servidor; a segundo permite

obter um mapa cujos parâmetros são bem definidos. A terceira operação é opcional e permite obter informação sobre dados que estão contidos no mapa. Estas operações podem ser realizadas através de um pedido *HTTP*. O pedido é realizado na forma de um *URL* - *Uniform Resource Locators*. O conteúdo do *URL* depende da operação que se pretende e indica qual é a informação que deve ser apresentada no mapa.

A informação geográfica contida num *WMS*, pode ser vista como um conjunto de camadas (*Layers*), que contém uma determinada informação. O protocolo descreve três operações, sendo uma delas opcional. De seguida descrevem-se estas operações.

### *GetCapabilities*

Esta operação permite obter informação sobre quais os dados existentes no servidor, nomeadamente, o tipo de mapa (*bitmap* ou *vectorial*), informação sobre área de visualização de imagem, o sistema de coordenadas, quais as camadas existentes, entre outras. Esta operação devolve um ficheiro *XML* com a informação especificada. A Tabela 3.1 descreve os parâmetros desta operação.

Exemplo de um pedido *GetCapabilities*: `http://localhost:8080/geoserver/wms?request=GetCapabilities`

Tabela 3.1: Parâmetros de um pedido *GetCapabilities*.

Parâmetros	Obrigatório /Opcional	Descrição
<i>VERSION=version</i>	Opcional	A versão do pedido
<i>SERVICE=WMS</i>	Obrigatório	O tipo do serviço
<i>REQUEST=GetCapabilities</i>	Obrigatório	O nome do pedido
<i>FORMAT=MIMET_type</i>	Opcional	O formato dos meta-dados obtidos
<i>UPDATESEQUENCE</i>	Opcional	Número de sequência para controlo da <i>cache</i>

### ***GetMap***

Esta operação permite obter um mapa no formato de imagem. Os pedidos definidos por esta operação, são constituídos por um conjunto de parâmetros que estão de acordo com a especificação definida pela *OGC*. Após o utilizador obter o ficheiro *XML* proveniente da operação *GetCapabilities*, pode usar a informação existente para realizar o pedido do mapa. O resultado é uma imagem contendo toda a informação requisitada no pedido. A Tabela 3.2 descreve os parâmetros associados a esta operação. Por exemplo, para obter um mapa com as seguintes características: Uma camada referente às estradas de nome *movisys:estradas*, a imagem no formato *png*, largura da imagem 240 pixeis e altura 290 pixeis, uma área definida pelos valores dados pela *bbox*, o pedido *GetMap* a efectuar era o seguinte:

```
http://localhost:8080/geoserver/wms?bbox=-150,50,-100,70&style=&Format=image/png&request=GetMap&layers=movisys:estradas&width=240&height=290&crs=EPSG:20790
```

### ***GetFeatureInfo***

O *WMS* pode opcionalmente permitir esta operação. Depois de efectuado o pedido *GetMap*, é possível obter informação adicional sobre dados particulares existentes num mapa. Nomeadamente estes dados referem-se a pontos de interesse existentes. A Tabela 3.3 descreve os parâmetros associados a esta operação. Por exemplo, para se obter informação sobre o ponto com as coordenadas *x= 159* pixeis e *y= 220* pixeis, referente à camada de dados *movisys:estradas*, em que a primeira parte do pedido, desde do endereço do servidor até ao tipo de pedido definido pelo parâmetro *REQUEST=GetFeatureInfo*, representam a imagem gerada que contém o ponto a obter informação, o pedido a efectuar era o seguinte:

```
http://localhost:8080/geoserver/wms?BBOX=506480.631578,4404173.889293,506875.808248,4404784.415047&LAYERS=movisys:estradas&STYLES=&HEIGHT=240&WIDTH=290&SRS=EPSG:20790&FORMAT=image/png&SERVICE=WMS&VERSION=1.1.1&REQUEST=GetFeatureInfo&EXCEPTION=&X=159&Y=220&INFO_FORMAT=text/html&QUERY_LAYERS=movisys:estradas&
```

Tabela 3.2: Parâmetros de um pedido *GetMap*.

Parâmetros	Obrigatório Opcional	Descrição
<i>VERSION=1.3.0</i>	Obrigatório	A versão do pedido
<i>SERVICE=WMS</i>	Obrigatório	O tipo do serviço
<i>REQUEST=GetMap</i>	Obrigatório	O nome do pedido
<i>LAYERS=layers_list</i>	Obrigatório	Uma lista contendo as camadas de dados separados por vírgula
<i>STYLES=styles_list</i>	Obrigatório	Uma lista com os vários estilos, um por cada camada
<i>CRS=namespace:identifier</i>	Obrigatório	Sistema de coordenadas dos dados
<i>BBOX=minx,miny,maxx,maxy</i>	Obrigatório	A área relativa ao pedido, nas unidades referidas pelo <i>CRS</i>
<i>WIDTH</i>	Obrigatório	Largura em pixels da imagem
<i>HEIGHT</i>	Obrigatório	Altura em pixels da imagem
<i>FORMAT=output_format</i>	Obrigatório	O tipo de imagem que se pretende, <i>Bitmap</i> ou <i>Vectorial</i>
<i>TRANSPARENT=TRUE—FALSE</i>	Opcional	A transparência de fundo do mapa
<i>BGCOLOR=color_value</i>	Opcional	A cor de fundo do mapa
<i>EXCEPTIONS=exception_format</i>	Opcional	O formato das exceções a retornar
<i>TIME=time</i>	Opcional	Indica um valor temporal para a obtenção de uma camada
<i>ELEVATION=elevation</i>	Opcional	Representa um nível de informação referente a uma camada

### 3.2.2 Web Feature Services

A especificação *WMS*, permite às aplicações cliente, obter mapas em forma de imagem, provenientes de vários servidores de mapas na *Internet*. De modo similar a especificação *WFS*, permite a um cliente obter e manipular dados geográficos especificados na linguagem *GML* (*Geography Markup Language*) e acedidos através de uma interface *HTTP*.

Tabela 3.3: Parâmetros de um pedido *GetFeatureInfo*.

Parâmetros	Obrigatório /Opcional	Descrição
<i>VERSION=1.3.0</i>	Obrigatório	A versão do pedido
<i>REQUEST=GetFeatureInfo</i>	Obrigatório	O nome do pedido
<i>map request part</i>	Obrigatório	Cópia parcial do pedido do mapa que gerou a informação pretendida
<i>QUERY_LAYERS=layer_list</i>	Obrigatório	Uma lista com as várias camadas a interrogar, separados por virgula
<i>INFO_FORMAT=output_format</i>	Obrigatório	O formato da informação a obter
<i>I=pixel_column</i>	Obrigatório	Coordenada I do ponto referente ao dado pretendido
<i>J=pixel_row</i>	Obrigatório	Coordenada J do ponto referente ao dado pretendido
<i>FEATURE_COUNT=number</i>	Opcional	O número de dados a retornar informação, (1 por defeito)
<i>EXCEPTIONS=exceptions_format</i>	Opcional	O formato das excepções a retornar pelo WMS, (XML por defeito)

Os dados geográficos possuem um estado que é definido por um conjunto de propriedades, em que cada propriedade pode ser visto como um tuplo (nome, tipo, valor). Por exemplo, uma determinada estrada pode conter uma propriedade de nome *size*, do tipo *float*, com o valor 3 km, pode conter outra propriedade de nome *type*, do tipo *string*, com o valor *highway*. Estas propriedades definem o estado deste dado geográfico.

Estes dados podem conter uma propriedade denominada de geometria. A geometria indica a sua representação num espaço dimensional e pode ser representado por pontos, linhas ou polígonos, neste caso uma estrada seria representada geometricamente por um conjunto de linhas. Segundo definições da *OGC* uma geometria pode conter uma colecção de outras geometrias, permitindo um agregado de propriedades.

Descreve-se de seguida o fluxo de informação processado entre uma aplicação cliente e um servidor *WFS*:

- A aplicação cliente realiza um pedido ao documento *capabilities* do *WFS*, de modo a obter a descrição de todas as operações que o mesmo

suporta e uma lista de todos os dados;

- A aplicação cliente opcionalmente realiza um pedido ao servidor de modo a obter informação sobre um ou mais tipos de dados existentes;
- Com base na definição dos dados obtidos, o cliente realiza um determinado pedido ao servidor. O servidor processa o pedido e devolve a resposta, que tipicamente indica o resultado do processamento. Contudo, no caso de devolver um erro, este é acompanhado da respectiva descrição.

O *WFS* descreve um conjunto de operações que podem ser requisitadas. Estas operações permitem inserir, actualizar, apagar e realizar interrogações sobre dados geográficos.

### ***GetCapabilities***

Este pedido descreve que operações existem e que tipo de dados são suportados pelo *WFS*.

Exemplo de um pedido *GetCapabilities*: `http://localhost:8080/geoserver/wfs?request=GetCapabilities`

### ***DescribeFeatureType***

Esta operação permite descrever a estrutura e o conteúdo dos dados existentes no servidor dos serviços. Por exemplo, para obter a descrição de uma camada de dados referente às estradas realiza-se o seguinte pedido:

`http://localhost:8080/wfs.cgi?SERVICE=WFS&VERSION=1.1.0&REQUEST=DescribeFeatureType&TYPENAME=movisys:estradas`

O *URL* contém o endereço do servidor e a respectiva interrogação definido pelo tipo do serviço (*WFS*), a versão (1.1.0), o tipo do pedido (*DescribeFeatureType*) e que dados se pretende a descrição (*movisys:estradas*).

### ***GetFeature***

Esta operação permite obter uma instância dos dados existentes, isto é um ficheiro *XML* contendo uma cópia integral de toda a informação que representa os dados requisitados. O cliente deve especificar que dados e que

propriedades pretende obter. Por exemplo, para obter toda a informação referente a uma estrada específica, a *IC19*, é necessário indicar o endereço do servidor e a respectiva interrogação. Os parâmetros do pedido incluem o nome do serviço (*WFS*), a versão (1.1.1), o pedido (*GetFeature*) e o identificador do dado que se pretende toda a informação (*estrada\_IC19*), o pedido a efectuar era o seguinte:

```
http://localhost:
8080/wfs.cgi&SERVICE=WFS&VERSION=1.1.0&REQUEST=
GetFeature&FEATUREID=estrada_IC19
```

### ***Transaction***

Uma transacção é composta por operações que permitem modificar dados existentes. Tais operações podem ser: criar, apagar e actualizar dados geográficos. Por exemplo, para se eliminar da base geográfica um tipo de dados denominado estrada *IC16* deve-se indicar para além do endereço do servidor a respectiva interrogação, incluindo o tipo do Serviço *WFS*, a versão (1.1.0), o tipo do pedido (*Transaction*), qual a operação pretendida (*Delete*) e o identificador do dado que se pretende remover (*estrada\_IC16*). O pedido a efectuar era o seguinte:

```
http://localhost:8080/wfs.cgi?SERVICE=
WFS&VERSION=1.1.0&REQUEST=Transaction&OPERATION=
Delete&FEATUREID=estrada_IC16
```

### ***LockFeature***

Esta operação permite bloquear o acesso a uma ou mais instâncias dos dados geográficos, para que não possam ser acedidos durante a realização de uma transacção. O objectivo é garantir a integridade dos dados. Por exemplo, para bloquear o acesso a uma camada de dados durante uma transacção é necessário indicar para além do endereço do servidor a respectiva interrogação, contendo o tipo de serviço (*WFS*), a versão (1.1.0), o pedido (*LockFeature*) e qual o nome da camada de dados que se pretende bloquear (*movisys:estradas*). O pedido a efectuar era o seguinte:

```
http://localhost:
8080/wfs.cgi&SERVICE=WFS&VERSION=1.1.0&REQUEST=
LockFeature&TYPENAME=movisys:estradas
```



## Parâmetros dos pedidos

Os parâmetros dos pedidos *WFS*, tal como os parâmetros do serviço *WMS*, são bem definidos pela sua especificação. Os parâmetros manipuláveis são: a versão do protocolo, o tipo de pedido, a área que envolve os dados (*Bounding box*), o tipo dos dados, etc. Os exemplos referidos anteriormente utilizam a maior parte desses parâmetros. A Tabela 3.4, descreve os parâmetros associados a estes pedidos.

Tabela 3.4: Parâmetros comuns de um pedido *WFS*.

Componentes do <i>URL</i>	Obrigatório /Opcional	Descrição
<i>http://servidor/caminho/pedido</i>	Obrigatório	O prefixo do pedido
<i>VERSION</i>	Obrigatório	A versão do pedido, (1.1.0 por defeito)
<i>SERVICE</i>	Obrigatório	O tipo do serviço, ( <i>WFS</i> por defeito)
<i>REQUEST</i>	Obrigatório	O nome do pedido
<i>NAMESPACE</i>	Opcional	Utilizado para especificar o espaço de nomes e seus prefixos
<i>Vendor-specific parameters</i>	Opcional	Opções adicionais do fornecedor (definidos no ficheiro <i>capabilities</i> )

### 3.2.3 *Styled Layer Descriptior*

A especificação *Styled Layer Descriptor* [27] está em conformidade com a norma *ISO 19105:2000 Geographic information* e define um mecanismo que permite aos utilizadores indicar o modo como os dados são apresentados no mapa. Permite ainda indicar a que dados geográficos são aplicados os estilos visuais. Desta forma existe um controlo sobre as referências visuais que são apresentados num mapa.

A especificação *WMS* permite às entidades que fornecem os serviços, definir um conjunto básico de opções de estilo. Estes estilos representam um conjunto de informações visuais para cada tipo de dados. Contudo, apesar de permitir ao utilizador escolher os estilos que pretende, apenas disponibiliza o nome dos estilos a utilizar. Por exemplo, não permite ao utilizador indicar qual a aparência visual dos dados no mapa. Para além

disso, o utilizador não pode definir os seus próprios estilos. A solução passa por utilizar uma linguagem que represente estes estilos e que tanto os clientes como os servidores possam compreender, o *Styled Layer Descriptor*.

Os dados geográficos propriamente ditos não possuem uma componente visual. Para que seja possível visualizar a informação geográfica num mapa, deve ser indicado qual o estilo visual que a representa. Existem duas maneiras de representar o estilo de uma camada de dados. A primeira opção é representar toda a informação da mesma maneira. Por exemplo, Uma camada que represente informação hidrográfica constituída por rios (linhas) e por lagos (polígonos), contém uma componente visual representada por uma cor de preenchimento a azul claro e a cor dos contornos a azul escuro. Este tipo de estilo não requer conhecimento dos atributos dos dados, pois independentemente de serem rios ou lagos a apresentação visual dos dados é realizada da mesma forma.

Uma outra aproximação é aplicar estilos definidos dependendo de algum atributo. Por exemplo, numa camada que represente estradas, as auto-estradas podem ser representadas por uma cor específica e possuir uma dada largura das linhas e as ruas são representadas por outra cor e diferente largura das linhas, de modo a diferenciá-los. Deste modo é necessário determinar quais os atributos que representam os diferentes tipos de estradas.

O *WMS* possui uma operação que permite obter uma descrição das camadas existentes no servidor de mapas, o *DescribeLayer*. Os atributos podem ser obtidos através da operação *DescribeFeatureType* do *WFS*.

A apresentação de um mapa pode ser realizada através da combinação de camadas e de estilos que definem essas camadas, recorrendo a uma determinada ordem. A especificação *WMS* fornece uma maneira de definir uma camada de estilo, combinando o nome de uma camada e o nome de um estilo. A aparência de um mapa pode ser definida usando a linguagem *SLD*. Existem três aproximações para uma aplicação poder representar os seus dados através do *SLD*:

1. A aplicação realiza um pedido do mapa e indica no *URL* a localização de um *SLD* remoto.

2. A aplicação realiza um pedido do mapa e inclui no *URL* o próprio *SLD*.
3. A aplicação realiza um pedido do mapa e não indica nenhum *SLD*. Neste caso são utilizados os *SLDs* pré definidos no servidor. Cada camada possui o seu estilo por omissão.

Caso uma aplicação utilize a ultima aproximação, o servidor utiliza o *SLD* por omissão, sabendo que cada camada possui obrigatoriamente um *SLD* por omissão. Caso uma aplicação opte pelo uso da primeira ou segunda opção, o *SLD* indicado possui preferência sobre o definido por omissão. A escolha da segunda opção pode conter problemas quando o *URL* do pedido é muito longo, pois os navegadores suportam um determinado limite de caracteres. Relativamente à primeira opção a localização do *SLD* tem de ser válida e tem de estar disponível.

Um documento *SLD* é definido por um conjunto de camadas de estilo. No topo hierárquico desse documento encontramos o elemento de nome *StyledLayerDescriptor*, que define a raiz do documento sobre o qual todos os outros elementos são definidos.

O elemento *NamedLayer* encontra-se logo a seguir à raiz do documento, representa uma camada que pode ser acedida através do seu nome e utilizado para definir a que camadas vão ser aplicados os estilos. A ordem de apresentação das camadas define a ordem pelo qual são visualizados no mapa.

Um *NamedLayer* pode ter associado vários estilos já existentes, denominados de *NamedStyle*. Estes estilos também podem ser definidos pelo utilizador, denominados de *UserStyle*. O elemento *NamedStyle* define o nome de um estilo existente por omissão, enquanto que o *UserStyle* define um estilo indicado por um utilizador ou aplicação.

O elemento *FeatureTypeStyle* que hierarquicamente se encontra a seguir ao elemento *UserStyle*, contém a informação que define as regras e os filtros ao qual os estilos são aplicados. Deste modo é possível definir quais os dados que são apresentados e como são apresentados. Por exemplo, a Figura 3.2 apresenta um ficheiro *SLD* referente a uma camada de dados

cujo nome (*NamedLayer*) é *movisis:estradas*. Este ficheiro possui um estilo (*UserStyle*), definido por duas regras (*Rule*). Estas regras baseiam-se na propriedade *Type*. Consoante o valor desta propriedade são apresentados num mapa as linhas correspondentes e respectiva cor e largura. Apenas os atributos especificados serão apresentados no mapa.

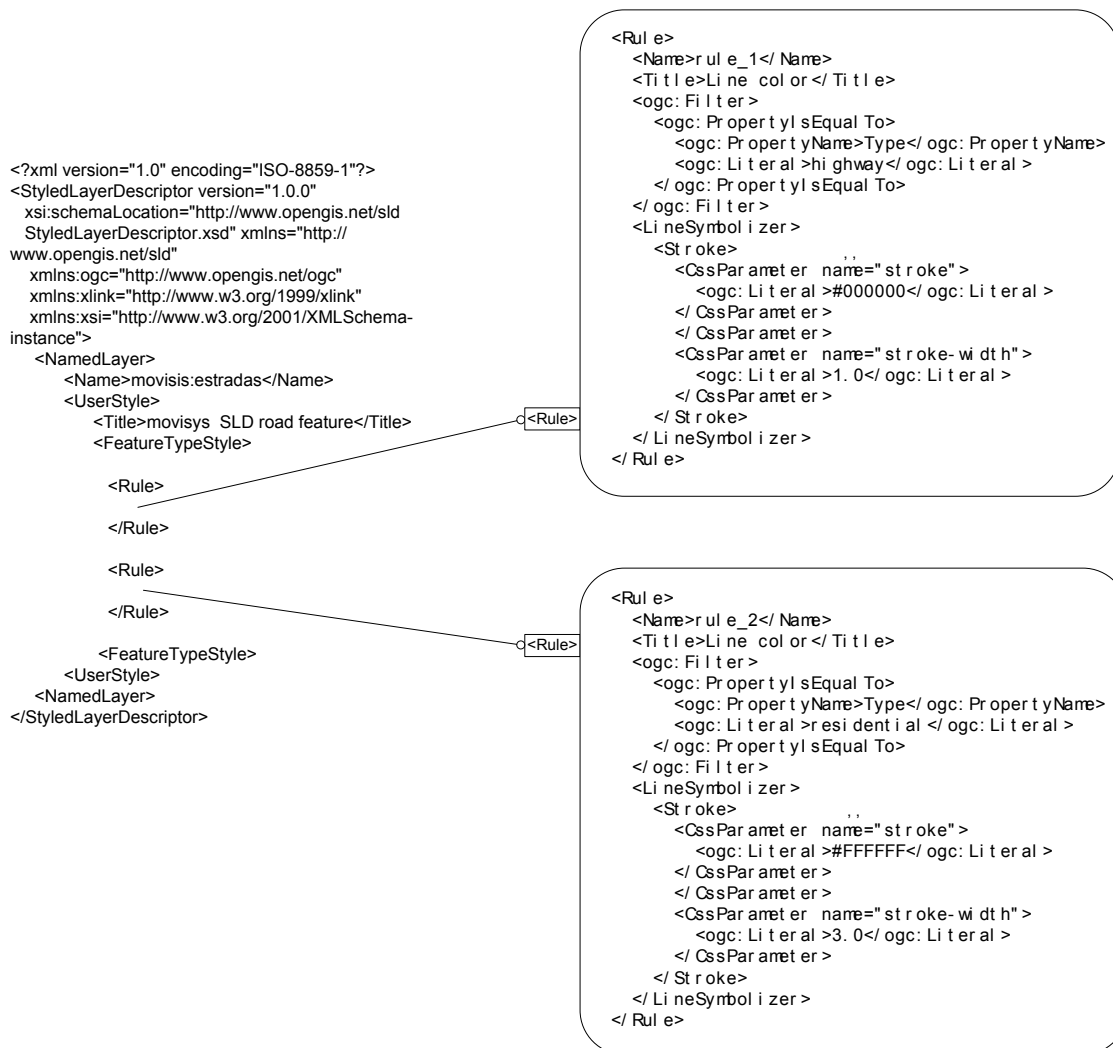


Figura 3.2: Ficheiro *SLD* definindo duas regras de estilo.

### 3.3 Servidores de Mapas

#### *Map Server*

O *MapServer* [14] é uma plataforma de domínio público, direccionada à publicação de dados geográficos e construção de aplicações sobre mapas na Web. Foi desenvolvido nos anos 90, na Universidade do *Minnesota*

e existe para diferentes plataformas, tais como: *Windows*, *Linux* e *Mac OS X*. A sua principal característica baseia-se na apresentação de dados espaciais, mapas, imagens, dados *bitmap* e vectoriais na *Internet*. O *MapServer* permite criar imagens de mapas geográficos. Dentro das principais características do *MapServer* citamos algumas como:

- Possui suporte a várias tecnologias normalizadas *OGC* orientadas à informação geográfica, tais como o *WMS*, *WFS*, *SLD*, *GML*;
- Suporte a várias plataformas e ambientes de programação tais como: *PHP*, *Python*, *Perl*, *Ruby*, *Java*, e *.NET*;
- Suporte às projecções sobre mapas, permite transformar sistema de coordenadas de um mapa em outro;
- Suporte a vários formatos de apresentação, *Vectorial* e *Bitmap* tais como: *ESRI shapefiles*, *PostGIS*, *Oracle Spatial*, *MySQL* e outros.

### ***Geoserver***

O *GeoServer* [5] é um servidor de mapas de domínio público, implementado na linguagem *Java* e disponível sobre a licença *GPL 2.0*, que permite aos utilizadores partilhar e editar dados geográficos na *Web*. A sua implementação é baseada nas tecnologias normalizadas *OGC* - *WMS*, *WFS* e *WCS* (*Web Coverage Service*). O objectivo deste servidor é facilitar o suporte às normas, de modo a garantir interoperabilidade entre as aplicações. Deste modo é possível a qualquer aplicação interagir com os serviços fornecidos pelo mesmo. As principais características do *GeoServer* são:

- Compatível com protocolos *Web Services* orientados a informação geográfica, tais como o *Web Map Services* e o *Web Feature Services*.
- O *WMS* fornece imagens no formato *JPEG*, *GIF*, *PNG*, *SVG* e *GML*;
- Suporte à manipulação de informação geográfica em repositórios de informação tais como, *PostGis*, *MySQL*, *Shapefile*, *ArcSDE* e *Oracle*;
- Suporte à linguagem *SLD*, quer para as definições do utilizador, quer para as definições de configuração.



## **Capítulo 4**

# **Informação Geográfica em Dispositivos Móveis**

A informação geográfica apresentada nos mapas, muitas vezes torna-se confusa, contendo informação que não interessa aos utilizadores durante a realização de uma determinada tarefa. Muitos sistemas de visualização de informação geográfica fornecem aos utilizadores informação consoante os níveis de aproximação ou afastamento. Contudo, não permitem filtrar parte dessa informação. No contexto dos dispositivos móveis esta tarefa torna-se ainda mais complicada devido às limitações impostas pelos mesmos.

Neste capítulo apresentam-se as soluções encontradas e os mecanismos utilizados que permitem aos utilizadores filtrar a informação geográfica a visualizar de acordo com as suas preferências, no contexto dos dispositivos móveis. Na secção 4.1 apresenta-se o enquadramento e o trabalho relacionado. Na secção 4.2 apresenta-se os aspectos relativos à fase de análise no que se refere ao trabalho desenvolvido. Na secção 4.3 abordam-se os aspectos que envolvem a fase de desenho. Por último é apresentada a secção relativa à implementação do protótipo, referindo as tecnologias utilizadas e alguns cenários de utilização que envolvem o protótipo desenvolvido.

### **4.1 Enquadramento e trabalho relacionado**

Este trabalho foi desenvolvido no âmbito de um projecto de investigação em que o objectivo é a visualização em dispositivos móveis de dados georeferenciados, organizados em categorias com múltiplos atributos que permite aos utilizadores filtrar a informação geográfica consoante as suas preferências.

Esta secção tem como objectivo enquadrar o trabalho relacionado, realizando uma descrição sobre o sistema *MoViSys*, desde a metodologia de concepção seguida, até à descrição das componentes de arquitectura, tendo em conta os trabalhos desenvolvidos em [46] e [37]. É apresentada também uma descrição do sistema *GiModig*, descrevendo a sua arquitectura e referindo qual o impacto do seu estudo no desenvolvimento deste projecto.

### **Sistema *MoViSys***

O sistema *MoViSys* é um sistema de visualização em dispositivos móveis para dados geo-referenciados, organizados em categorias com múltiplos atributos.

Uma vez que o ecrã dos dispositivos móveis tem dimensões muito reduzidas, a representação de todos os pontos de interesse que satisfazem uma determinada pesquisa, originam frequentemente imagens confusas. Deste modo, para que a informação seja compreensível para o utilizador, é necessário controlar o número de símbolos a desenhar no ecrã. Para que tal seja possível, é necessário reduzir o número de símbolos sempre que existam demasiados pontos de interesse ou se verifique demasiada proximidade entre eles. No sistema desenvolvido, é utilizado uma combinação de mecanismos de filtragem, que permitem eliminar resultados menos relevantes, e operadores de generalização (agregados), que permitem agrupar resultados próximos. Para filtrar os pontos de interesse a mostrar ao utilizador, utilizou-se uma função que permite quantificar o grau de interesse que o utilizador possui num determinado ponto [46]. Nas situações em que não foi útil utilizar os agregados, foi utilizado um operador de afastamento que permitiu reduzir a área dos símbolos que está sobreposta, tornando a imagem menos confusa.

A interface do sistema *MoViSys* proporciona ao utilizador a selecção de tópicos de interesse bem como a parametrização da função de grau de interesse. Os utilizadores especificam a pesquisa impondo restrições aos atributos das categorias seleccionadas a fim de visualizar os resultados destas no mapa área geográfica considerada.

A concepção do sistema *MoViSys* foi realizado de acordo com uma metodologia centrada no utilizador. A fase de análise de requisitos envolveu entrevistas informais com potenciais utilizadores, simulações de sessões e



o estudo de diversos projectos com objectivos comuns ao sistema *MoViSys*. O resultado deste processo permitiu identificar as funcionalidades requeridas e as propriedades e comportamento da interface numa fase inicial. Na fase de prototipagem foram realizadas e avaliadas algumas alternativas de simbologia e *design* da interface, de especificação de pesquisa, que permitiram fundamentar algumas decisões, nomeadamente sobre que símbolos mais adequados e os tipos de *widget* a incorporar. O protótipo tem sido iterativamente refinado de modo a concretizar a totalidade das funcionalidades.

A Figura 4.1 apresenta o diagrama de classes em *UML* do modelo de dados concebido para a representação das características das categorias, dos símbolos e dos pontos de interesse. Este modelo de dados contém a classe *Categoria* onde são guardados os nomes das categorias existentes; na classe *Atributos* estão guardados os nomes dos atributos e seu tipo. A classe *API* define os valores máximos e mínimos de cada atributo de cada categoria, como por exemplo, o preço de um Hotel. A classe *PontoDeInteresse* contém o seu identificador único, o nome e a latitude e a longitude da localização do ponto de interesse. Por sua vez, a classe *Característica* guarda todas as suas características específicas de uma categoria. Por exemplo, se a categoria for “área de serviço”, então as suas características são: o tipo de gasolina, a marca, o horário de funcionamento, os extras que esta contém, a localidade e o número de telefone. A classe *Símbolos* contém o seu identificador e o respectivo nome do ficheiro que permite identificar qual o ícone correcto para o ponto de interesse encontrado. A classe *Símbolos* tem como classes derivadas a classe *Simples*, para representar um ponto de interesse e a classe *Agregado* para representar pontos de interesse agregados.

A Figura 4.2 apresenta a arquitectura global do protótipo do sistema *MoViSys*. O protótipo foi desenvolvido para *Pocket PC*, com sistema operativo *Windows Mobile 5.0*, utilizando o *.NET Compact Framework*. Os pontos de interesse são guardados numa base de dados *SQL Server* e os mapas são obtidos através do *Google Maps WebServer*.

Nesta arquitectura sempre que o utilizador altera as coordenadas da área que está a visualizar, ou altera a ampliação do mapa, o Cliente de *HTTP*, envia um pedido ao servidor *Web* do *Google Maps*, que lhe devolve uma nova imagem do mapa para as coordenadas e nível de ampliação desejado.

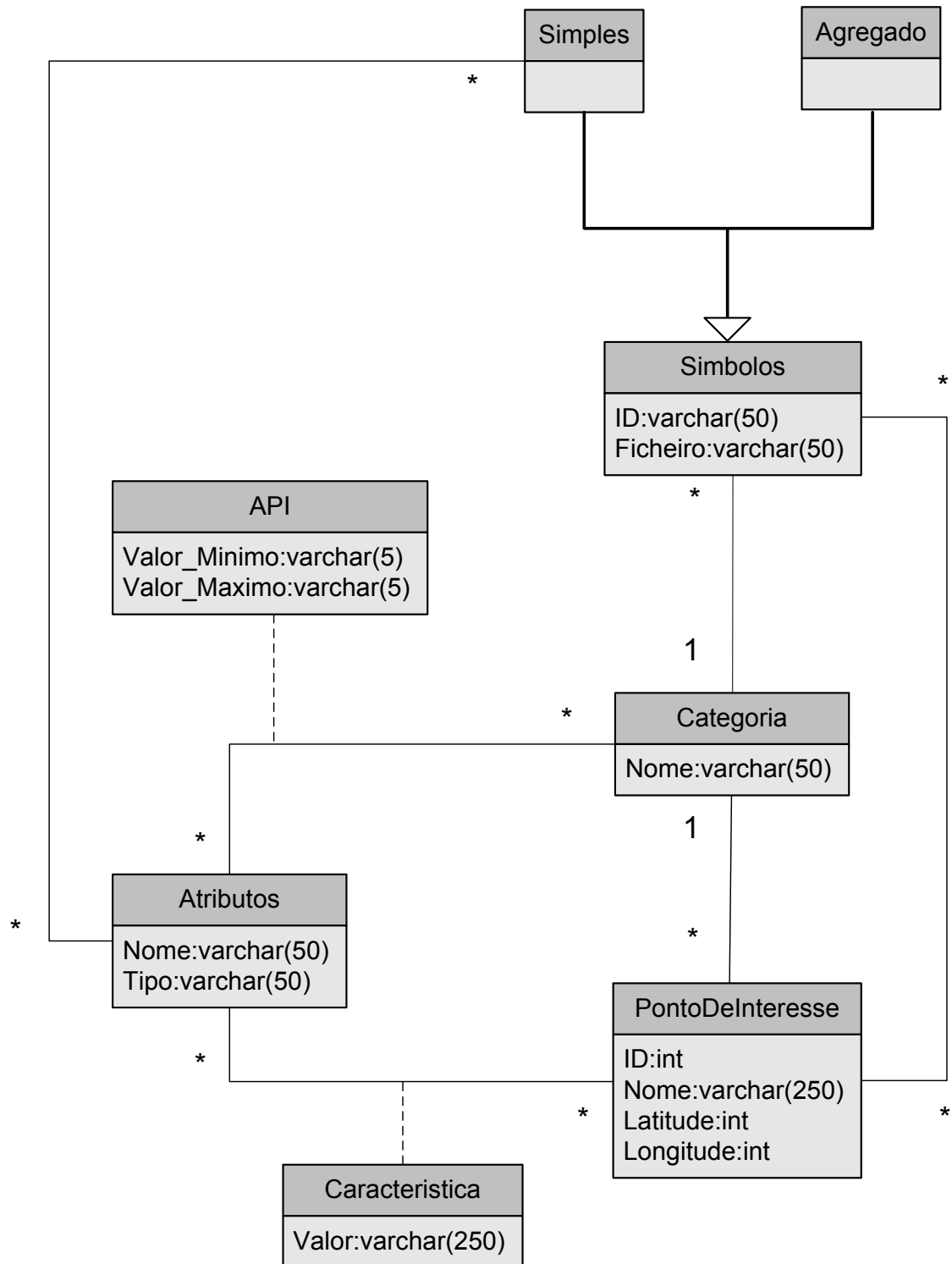


Figura 4.1: Modelo de dados base do sistema MoViSys.

Do mesmo modo sempre que há uma alteração da área visível, ou da pesquisa feita pelo utilizador, é enviado um pedido para o servidor *SQL Server*, que devolve para o Cliente *SQL Server* a lista de pontos de interesse

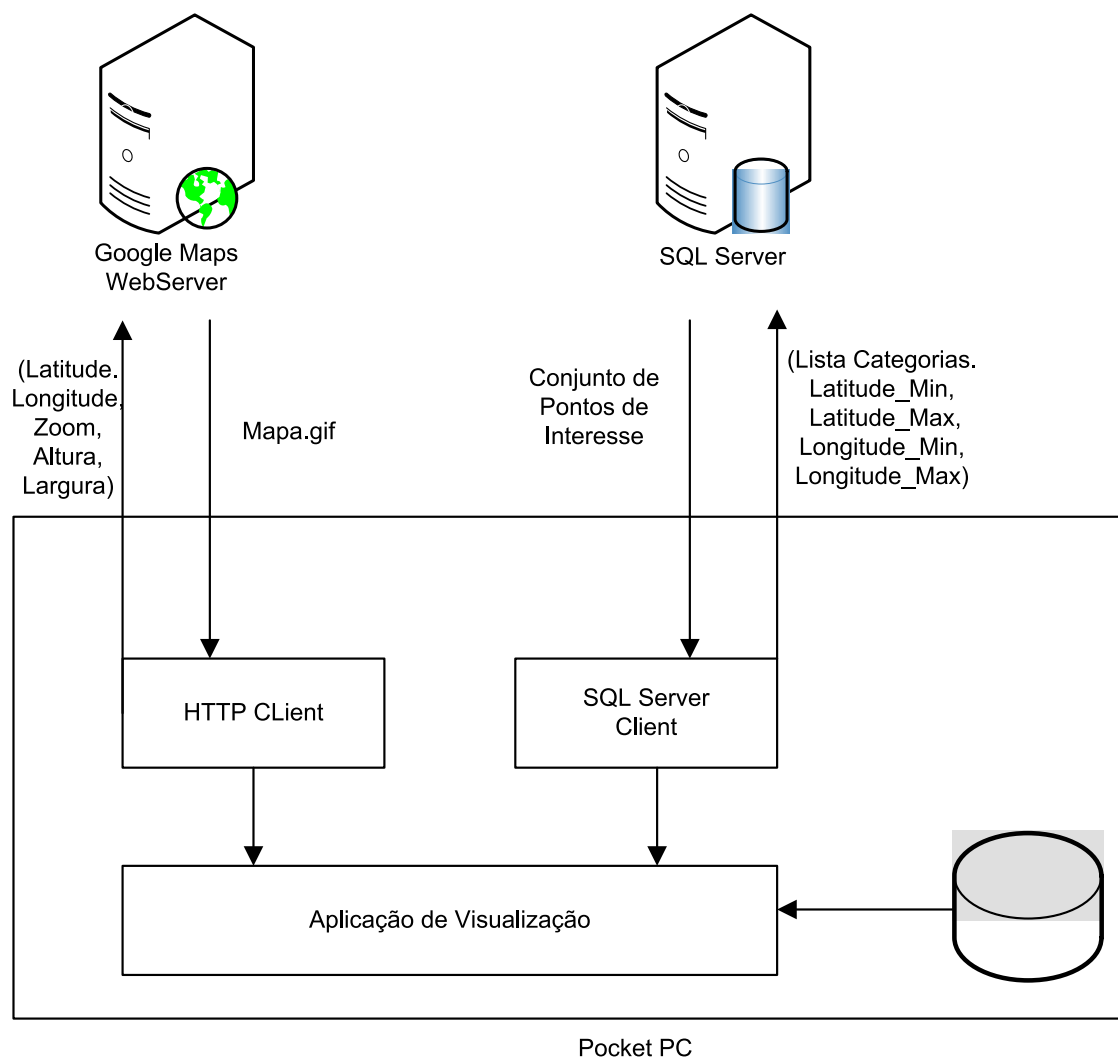


Figura 4.2: Arquitectura do sistema MoViSys.

visíveis, e os respectivos atributos. De notar, que a interface é dinamicamente construída, baseando-se nas definições existentes no modelo de dados concretizado com uma base de dados *SQL Server*.

Devido a algumas limitações existentes no sistema *MoViSys*, um dos requisitos propostos neste trabalho é a exploração de formatos vectoriais de modo a permitir a manipulação da informação visualizada nos mapas. Numa primeira abordagem explorou-se o formato vectorial *SVG*. Contudo, esta abordagem foi abandonada na medida em que implicava realizar um interpretador capaz de descodificar o conteúdo do ficheiro *XML* obtido pelo servidor de mapas, realizar o desenho do mapa e apresentar a imagem. O grau de complexidade da primeira opção levou à escolha de uma segunda opção mais simples. A escolha passou por obter as imagens em formato

*bitmap*, através de pedidos *HTTP* ao servidor de mapas. Apesar da qualidade das imagens vectoriais se demonstrar superior, esta solução revelou ser mais simples para o desenvolvimento dos objectivos deste projecto. A manipulação da informação realizou-se através da configuração de dados vectoriais, *shapefiles*, do lado do servidor.

### Sistemas *GiModig*

Um dos principais sistemas estudados foi o sistema *GiModig*. Este sistema proporciona aos utilizadores métodos através dos quais podem aceder a dados geográficos em tempo real, a partir de dispositivos móveis [9]. O objectivo deste sistema foi a construção de uma infra-estrutura que fornecesse serviços de acesso, através de uma interface comum, a um conjunto de bases de dados geográficas.

Este sistema utiliza as especificações *WMS* e *WFS* na implementação dos seus serviços. A arquitectura deste sistema é baseado num conjunto de camadas de serviços em forma de pilha. Um serviço realiza os pedidos ao serviço da camada abaixo, este recebe a resposta e realiza algum processamento, sendo o resultado retornado ao serviço acima. Cinco camadas são propostas neste projecto:

- Camada 1 - Camada de dados. Espaço de dados codificados em formato *XML*;
- Camada 2 - Camada de Integração. Esta camada é responsável por converter o sistema de coordenadas para uma referência comum e por converter o esquema de dados recebidos para o esquema do sistema *GiModig*;
- Camada 3 - Camada processamento de dados. Nesta camada ocorre a generalização dos dados e é responsável por vários tipos de processamento espacial, que ocorrem antes da visualização do mapa. Processos de agregação e integração de dados cartográficos de origens diferentes são tidos em conta nesta fase;
- Camada 4 - Portal. Responsável por tratar os pedidos do cliente, encaminhar os pedidos para a aplicação *GiModig* e transformação do mapa. A abstracção desta camada permite o suporte de plataformas para os mais variados clientes como: *PCs*, *PDA*s e outros telefones móveis, através do uso de protocolos específicos;

- Camada 5 - Aplicação cliente. Disponibiliza os resultados e realiza os pedidos.

O estudo do sistema *GiModig* foi bastante importante para o desenvolvimento deste projecto, pois foi através deste estudo que se retirou as ideias principais para o desenvolvimento da arquitectura do protótipo desenvolvido, tal como o uso das tecnologias utilizadas (*WMS*, *WFS*).

## 4.2 Processo de Desenvolvimento

Nesta secção realiza-se uma breve descrição das etapas percorridas no desenvolvimento deste projecto desde da fase inicial até ao desenvolvimento do protótipo.

O processo de desenvolvimento utilizado neste projecto, baseou-se num modelo evolutivo. Inicialmente obteve-se um levantamento dos conceitos que englobaram o desenvolvimento deste projecto permitindo um enquadramento com o sistema *MoViSys*. Após esta fase foi realizada a análise dos requisitos realizando um estudo sobre o modelo de dados a utilizar, quais as funcionalidades propostas pelo sistema a desenvolver e a estrutura da interface. Após este estudo passou-se para a fase de desenho onde foi abordado a arquitectura do sistema e suas componentes baseando-se no estudo e análise de alguns sistemas como o *GiModig*. De seguida desenvolveu-se um protótipo inicial. Durante esta fase foram realizadas várias iterações, permitindo um refinamento progressivo do sistema. Durante a evolução do projecto existiram várias reuniões e foram especificados novos requisitos. À medida que foi existindo retorno por parte da equipa foi necessário incluir novas funcionalidades e melhorias.

Com este modelo os utilizadores podem obter rapidamente funcionalidades do sistema e verificar se estão em conformidade com o proposto. Sempre que surgir retorno por parte dos mesmos, melhorias vão sendo implementadas, até à versão final. A Figura 4.3 representa as etapas do modelo evolutivo utilizado na concretização deste projecto.

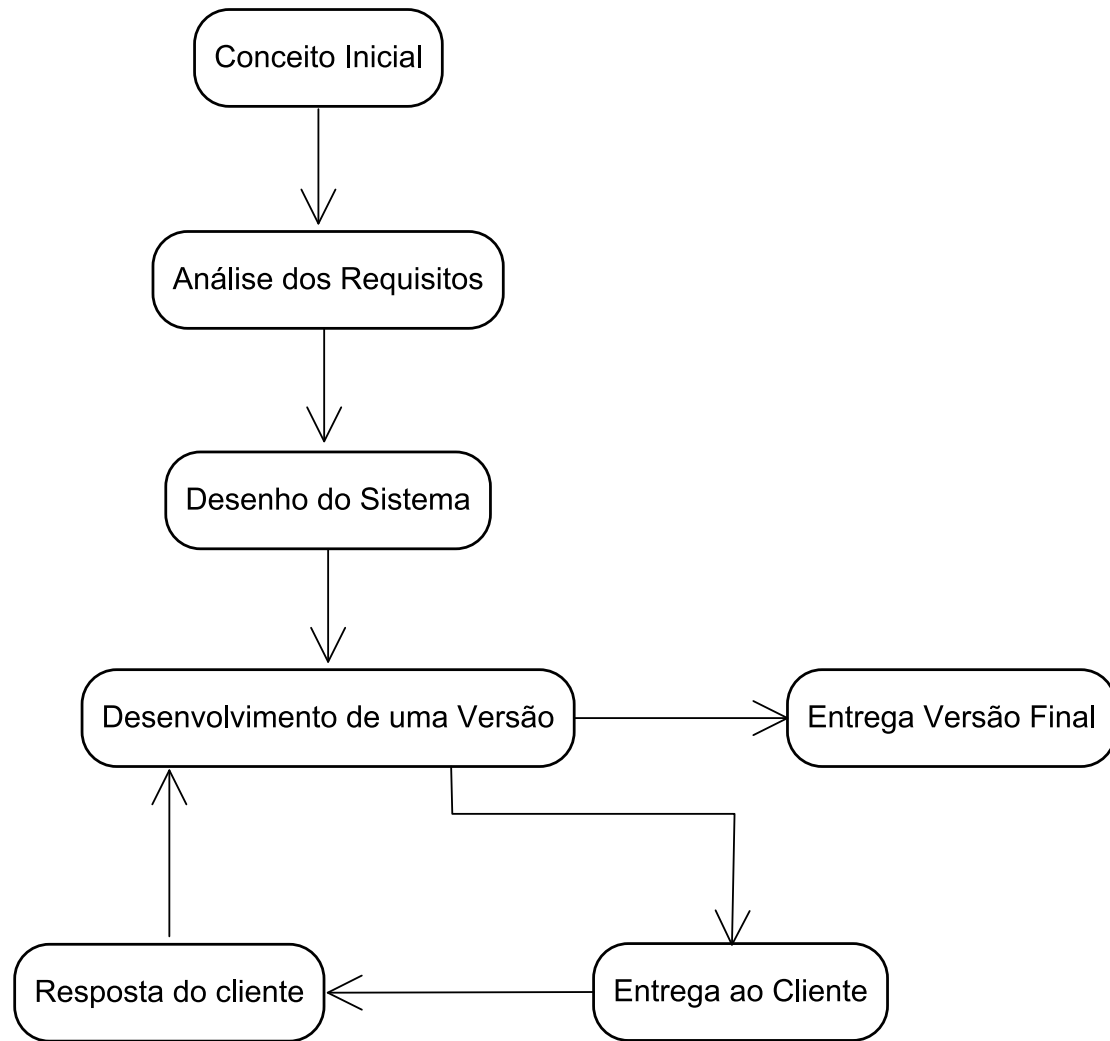


Figura 4.3: Modelo evolutivo referente ao processo desenvolvimento do projecto.

## 4.3 Análise

Nesta secção será apresentada a etapa de análise subjacente ao trabalho desenvolvido, referindo o modelo de dados utilizado como as operações de visualização implementadas e o desenho da interface.

### 4.3.1 Modelo Dados

O modelo de dados descreve um conjunto de componentes que representam as estruturas de dados que modelam o sistema, referindo os seus tipos e como se relacionam. Além disso existe um conjunto de operadores que manipulam essas estruturas e um conjunto de regras que definem que dados são válidos.

Para implementação do sistema desenvolvido, utilizou-se dois tipos de estrutura de armazenamento de dados, tabelas *PostgreSQL* com extensão *PostGIS*, para representar uma camada de dados referentes a pontos de interesse e *shapefiles* para representar as outras camadas.

A base de dados *PostgreSQL-PostGis* contém informação sobre qual a geometria dos objectos que representa. Contém também o sistema de coordenadas responsável pela projecção desses objectos. A Figura 4.4 refere o modelo de dados dessa camada.

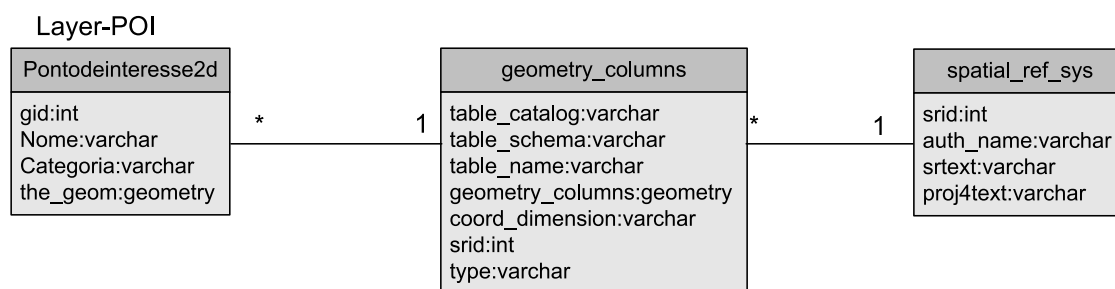


Figura 4.4: Modelo de dados representando a camada pontos de interesse.

Os dados referentes às *shapefiles* foram configurados directamente no servidor. A junção destas estruturas define o modelo de dados manipulados no servidor de mapas, o *GeoServer*. As operações de acesso a estas estruturas são executadas pelo mesmo, aquando dos pedidos aos seus serviços. A Figura 4.5 representa o modelo de dados global utilizado, contendo todas as camadas de dados de que modelam o sistema desenvolvido.

Apesar dos dados possuírem origens diferentes, estes encontram-se no mesmo sistema de coordenadas, garantindo que as camadas de informação possam ser observadas no mesmo mapa numa dada área.

De modo a compreender melhor o funcionamento deste tipo de estruturas de dados e no que consiste uma base de dados geográfica, é realizado uma descrição da mesma. O objectivo é contextualizar este tipo de informação dentro do trabalho realizado.

## Sistemas de Gestão de Base de Dados Geográfica

Um sistema de base de dados geográfica é um sistema base de dados optimizada para gerir e guardar dados relacionados com objectos geográficos,

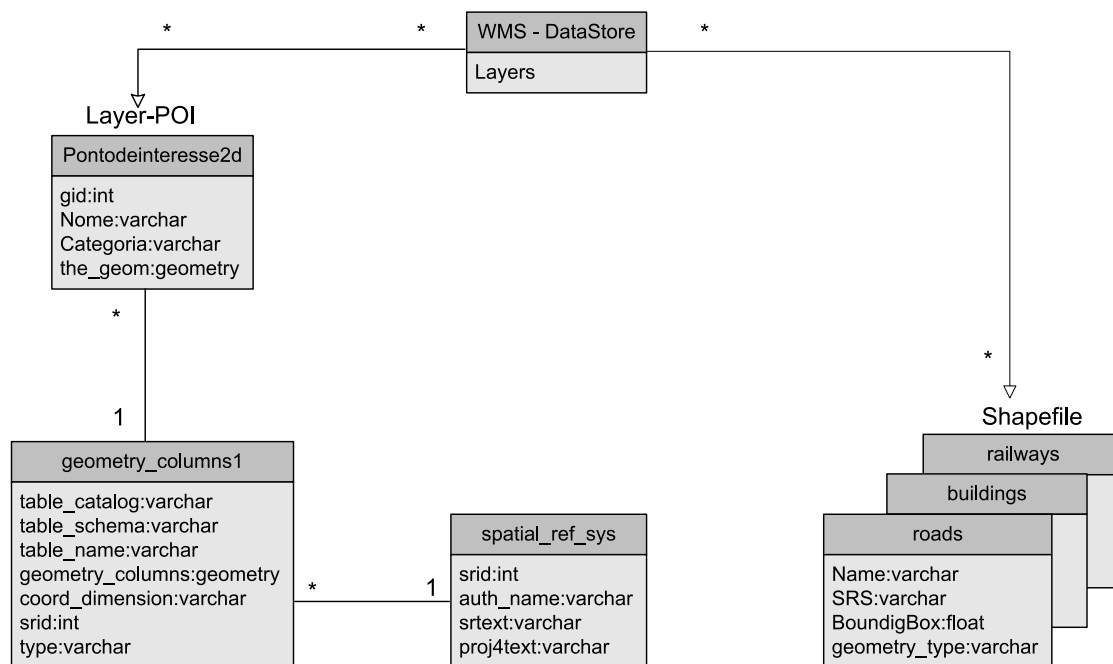


Figura 4.5: Modelo de dados representando a camada pontos de interesse e as restantes camadas referentes às *shapefiles*.

incluindo pontos, linhas e polígonos. Esta base de dados permite o acesso a diferentes tipos de dados. Contudo, para processar os dados geográficos é necessário um conjunto adicional de funcionalidades, denominadas de geometrias ou *features*. O *OGC*, definiu uma especificação denominada de *Simple Feature Specification*, para permitir adicionar funcionalidades geográficas aos sistemas de bases de dados deste tipo [1]. Alguns dos sistemas mais conhecidos são: *Oracle*, *MySQL* e *PostgreSQL* com extensão *PostGIS*. Esta extensão permite implementar para as diferentes geometrias, funções que permitem manipular os dados.

A especificação *OGC - Simple Feature Specification*, define um esquema *SQL* que suporta o armazenamento, consulta e actualização de colecções de dados geográficos simples, utilizando a *API* do *ODBC* (*Open Database Connectivity*) [28].

Os dados referidos possuem simultaneamente atributos geográficos e não geográficos. Os atributos geográficos simples possuem geometria 2D. Estes dados são armazenadas em bases de dados relacionais. As colecções de dados deste tipo são guardadas em tabelas que possuem uma coluna com valor geométrico, denominadas de *geometry column*.



### ***PostgreSQL-PostGIS***

O *PostgreSQL* [18] é um sistema de base de dados relacional e de domínio público que suporta grande parte da especificação *SQL*. O *PostGIS* é uma extensão da base de dados *PostgreSQL* e permite que seja usado no tratamento de bases de dados geográficos e aplicações que fornecem serviços de mapas. Este sistema de domínio público funciona de acordo com a licença *GNU GPL*. Esta extensão suporta um conjunto de operações que permite interrogações aos dados geográficos e permite garantia de integridade dos dados, de modo a que os dados sejam escritos nas tabelas sem contenção de recursos [17].

A especificação *Simple Features Specification* descreve o esquema utilizado para guardar a informação. À semelhança dos outros sistemas, o *PostGis* possui um conjunto de tabelas adicionais, com o objectivo de manipular a informação geográfica. A tabela *GEOMETRIC\_COLUMNS* define uma coluna para representação de uma geometria. Contém ainda o tipo de geometria, o sistema de coordenadas, a dimensão do objecto (2D) e um identificador, o *SRID*. A informação sobre cada sistema de coordenadas utilizado na base de dados encontra-se armazenado na tabela *SPATIAL\_REFERENCE\_SYSTEM*. Esta tabela contém um identificador (*SRID*), a autoridade que mantém o sistema de referência, o nome do sistema de referência e a sua respectiva descrição.

As tabelas de dados, para além de um identificador e os atributos, possuem uma chave estrangeira para a tabela que representa a sua geometria. Esta por sua vez, possui uma chave estrangeira indicando o sistema de coordenadas utilizado na representação do objecto. Esta informação é mais tarde utilizado na projecção dos dados sobre um mapa 2D.

#### **4.3.2 Funcionalidades**

De modo a permitir ao utilizador realizar as suas tarefas de modo eficaz é necessário compreender e definir um conjunto de operações que representem a manipulação dos dados que se pretende visualizar. Nesta secção define-se de modo conceptual quais as operações necessárias para o desenvolvimento do protótipo.

Um dos requisitos funcionais do sistema é a visualização num mapa de dados geográficos. Para além desta funcionalidade de base, pretende-se ainda que o utilizador defina as camadas de informação a visualizar, como por exemplo, seleccionar apenas algumas estradas principais e alguns pontos de interesse.

Deve existir uma outra funcionalidade que permita ao utilizador decidir visualizar pontos de interesse, representados pelos respectivos ícones, através da escolha de determinado atributo. Além disso o utilizador deve ter a possibilidade de obter informação sobre os mesmos.

Por fim, o utilizador deve ter a capacidade de visualizar a informação detalhada, ou simplesmente percorrer outras áreas de um mapa. Essas operações definem modos de afastamento, aproximação ou movimento do mapa, (*zoom*, *pan*, *move*). O diagrama de casos de uso que representa as funcionalidades, está representado na Figura 4.6.

### 4.3.3 Desenho da Interface

O objectivo de uma interface é facilitar a interacção entre o utilizador e um dado sistema, de uma forma simples e eficiente. Os processos de desenho devem combinar técnicas de funcionalidade com os elementos visuais de modo a criar um sistema operacional e adaptável.

O processo de desenho da interface requer uma análise dos requisitos que constituem as funcionalidade do sistema, de modo a concretizar os objectivos propostos e requer uma análise dos potenciais utilizadores. A análise do fluxo da informação deve ser acompanhada dos aspectos gráficos e dos testes de usabilidade. Contudo, apesar de não se ter seguido este modelo ao rigor, a modelação da interface do sistema foi progredindo sucessivamente durante as reuniões obtidas no desenvolvimento do projecto. Foi gradualmente analisada a aparência da interface, especificados os requisitos de um utilizador e realizados comparações com outros sistemas. Uma vez que o foco deste projecto não era a interface do protótipo foi escolhido um modelo menos rigoroso englobando análise, concepção e desenvolvimento de um protótipo. Contudo, sem a respectiva avaliação em conjunto com os testes de usabilidade.

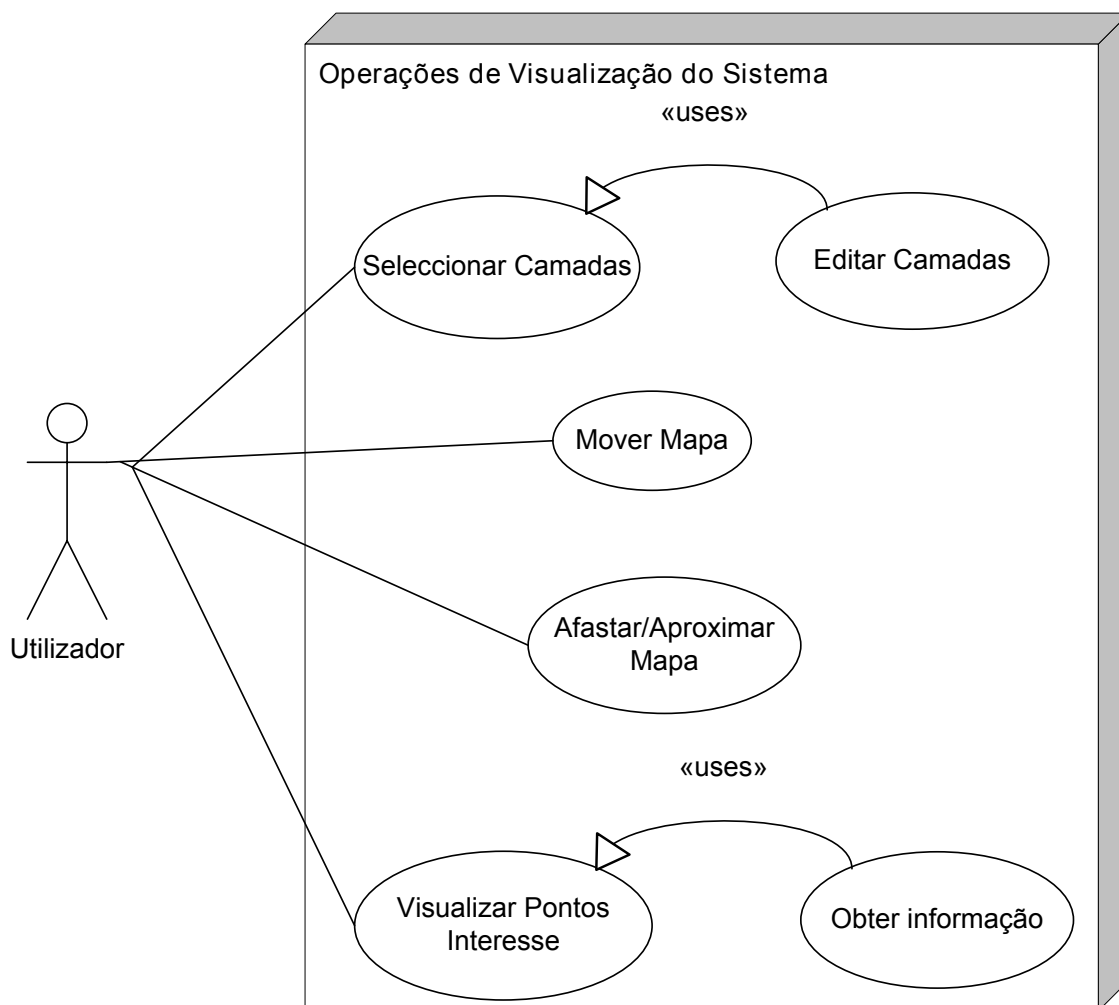


Figura 4.6: Diagrama Casos de Uso indicando operações de visualização.

Ao iniciar a aplicação cliente, é estabelecida a comunicação com o servidor e obtido um ficheiro *XML* com a informação existente: camadas de dados, os sistemas de coordenadas, os formatos de imagem, entre outros. Esta informação é carregada e é apresentado ao cliente através da interface principal, onde é disponibilizado uma camada de informação que representa os contornos de uma superfície, sobre a qual todas as outras camadas de informação vão assentar.

O utilizador neste momento pode adicionar camadas de informação e realizar operações de aproximação afastamento. No caso de o utilizador seleccionar alguma propriedade de uma camada de informação, por exemplo, largura de uma estrada, é activado um serviço que permite gerar um ficheiro *SLD* contendo essa informação. Caso contrário, toda a informação correspondente à mesma é visualizada. Relativamente aos pontos de in-

teresse, apesar de serem também uma camada de dados, quando seleccionados, a informação sobre os mesmos é gerado através do serviço que constrói o ficheiro *SLD*. O ficheiro *SLD* pode conter informação sobre pontos de interesse e/ou sobre as restantes camadas de informação e as propriedades a visualizar.

No caso das operações de afastamento, aproximação e movimento do mapa, o que sucede é uma alteração da área de visualização do mapa. Dependendo da opção seleccionada uma nova área é gerada utilizando uma transformação janela enquadramento. A nova área é enviada como parâmetro ao servidor no pedido do mapa. A Figura 4.7 representa o Diagrama de Actividades que traduz o fluxo de acções entre o utilizador a interface e o sistema.

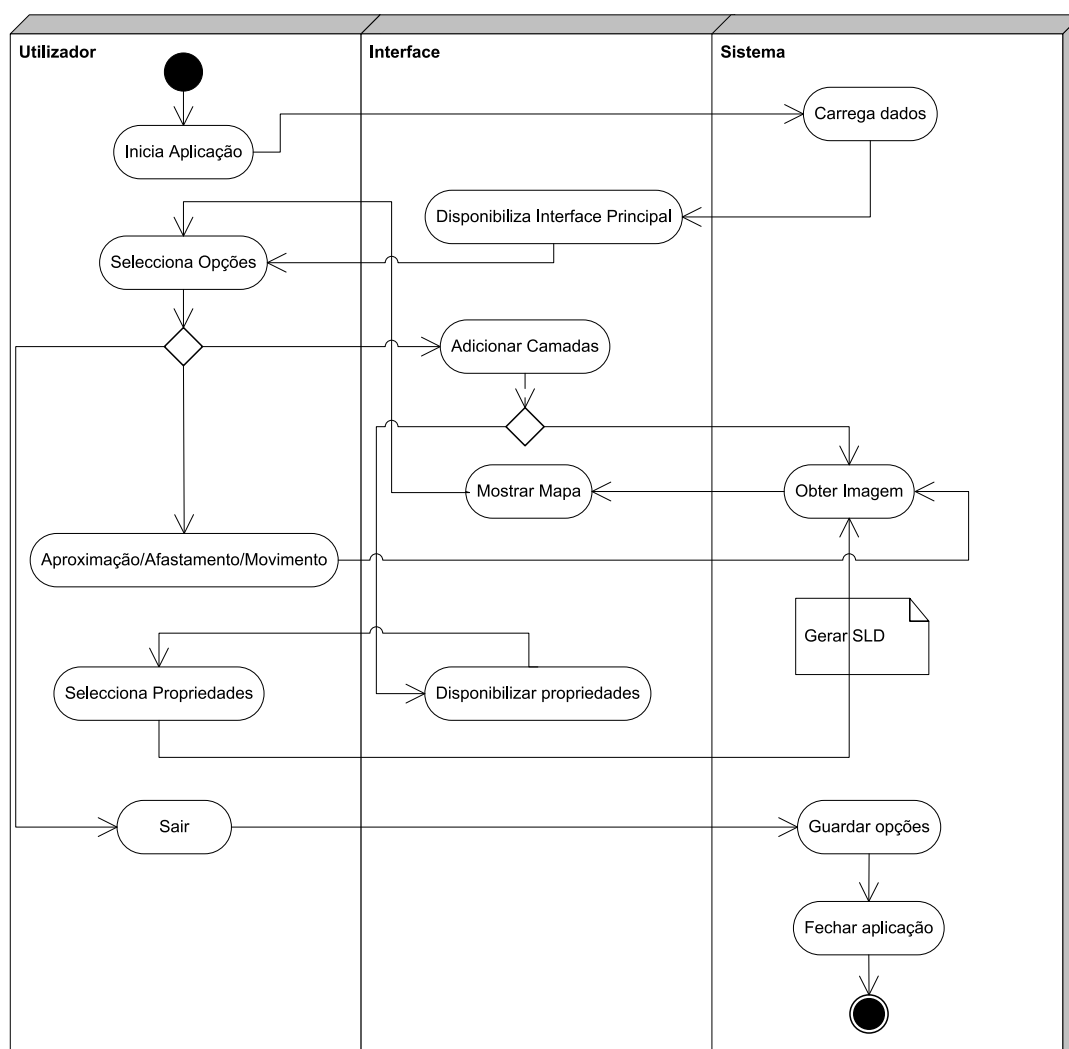


Figura 4.7: Diagrama Actividades representando as interacções entre a interface, o utilizador e o sistema.

## 4.4 Desenho da arquitectura

Na primeira parte desta secção descreve-se o modelo arquitectural e as suas respectivas características e componentes. Numa segunda parte apresenta-se o fluxo de informação entre as componentes.

As componentes que caracterizam a arquitectura do protótipo desenvolvido, seguem os modelos definidos pela *OGC*, de acordo com os protocolos *Web Services* orientados à informação geográfica, tais como o *Web Map Services*, *Web Feature Services* e a linguagem *Styled Layer Descriptor*.

A Figura 4.8 representa o modelo de arquitectura do sistema. A arquitectura do sistema é composta por três componentes essenciais, uma aplicação cliente, um servidor de mapas e um *Web Service* adicional. A comunicação entre estas componentes é realizada através de pedidos *HTTP* sobre uma rede, a *Internet*.

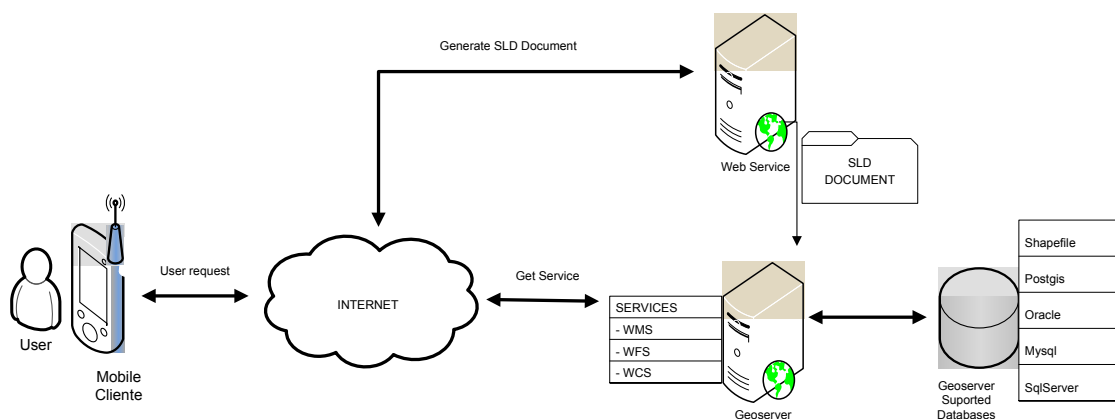


Figura 4.8: Modelo da Arquitectura do Sistema.

A aplicação cliente realiza os pedidos necessários para a visualização da informação. O servidor de mapas tem como função responder aos serviços requisitados pela aplicação cliente. O serviço adicional referido é baseado nos modelos *Web Service*. Este serviço é responsável por gerar em tempo real um ficheiro *XML* que contém informação relativa às camadas de dados pretendidas pelo utilizador. Este serviço apenas é activado quando seleccionado pelo utilizador, caso contrário as preferências por omissão são utilizadas.

De seguida descreve-se a interacção entre as componentes do sistema referindo as funcionalidades descritas anteriormente.

O fluxo da informação descreve a sequência de operações necessária para obtenção e visualização da informação. Quando um cliente realiza um pedido, a aplicação é responsável por processar o pedido e redireccionar o pedido para o servidor de serviços, o *GeoServer*. O *GeoServer* interpreta o pedido e gera uma resposta para a aplicação cliente. O resultado é disponibilizado ao cliente através da interface de visualização.

A Figura 4.9 descreve a fase inicial para a obtenção e visualização da informação geográfica através de um diagrama de sequência *UML*. Inicialmente a aplicação cliente começa por realizar um pedido *GetCapabilities*, para obter um ficheiro *XML* com os dados existentes no servidor. A aplicação cliente processa este ficheiro de modo a retirar os dados necessários para realizar as operações seguintes.

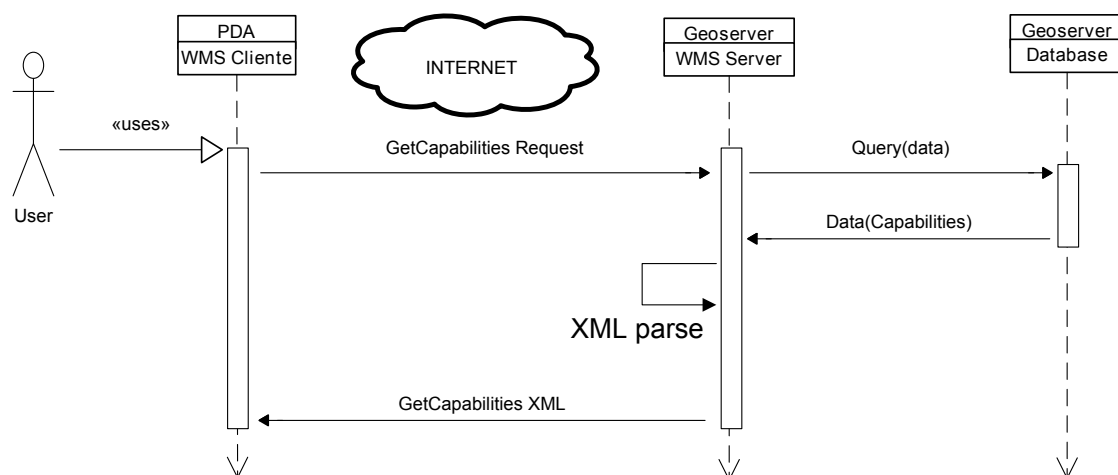


Figura 4.9: Fluxo de informação de um pedido *GetCapabilities*.

Após a informação sobre os dados existentes no servidor ter sido obtida o cliente pode seleccionar as camadas de dados que pretende visualizar e para cada uma delas, pode escolher as propriedades que pretende ver representadas no mapa. Caso o utilizador seleccione esta opção, é invocado um serviço para gerar um ficheiro *XML*, o *SLD*, contendo essa informação. É retornada a localização desse ficheiro. Após o ficheiro ter sido construído, é realizado um pedido *GetMap* ao servidor de mapas indicando a localização do ficheiro.

O *SLD* é interpretado pelo servidor para gerar uma imagem de acordo com as preferências definidas e sobrepõe-se as preferências geradas por omissão no servidor. A imagem resultante é retornada ao cliente e apresentada através da interface de visualização. A figura 4.10, descreve o fluxo de informação associado a uma operação de selecção de informação geográfica.

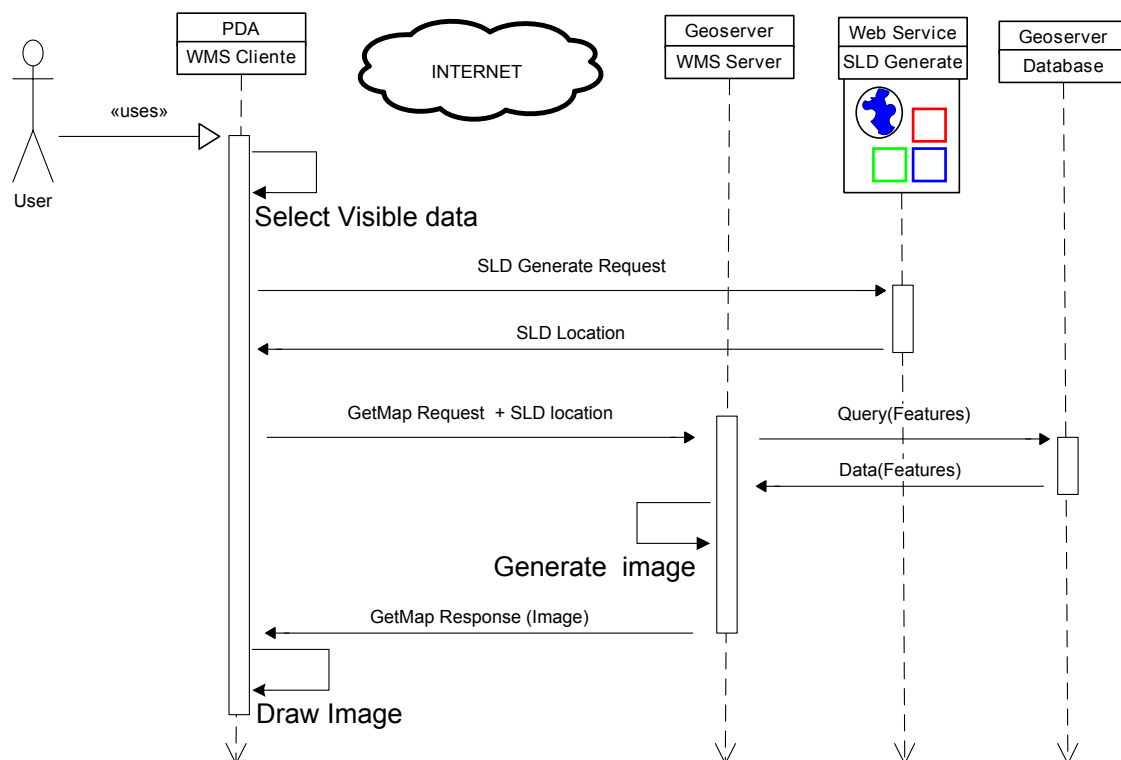


Figura 4.10: Fluxo de informação indicando pedido do mapa.

O utilizador pode também decidir realizar um movimento no mapa de modo a se deslocar para outras áreas, ou até mesmo obter uma maior aproximação ou afastamento dessas áreas. Neste caso é gerado um pedido do mapa com a informação da nova área a visualizar, o servidor gera a imagem correspondente e envia-a para a aplicação cliente, que a apresenta ao utilizador através da interface de visualização. A Figura 4.11 representa o fluxo de informação associado a estas duas funcionalidades.

## 4.5 Implementação

Na primeira parte desta secção são descritas as tecnologias utilizadas no desenvolvimento deste projecto. Na segunda parte é apresentado o protótipo

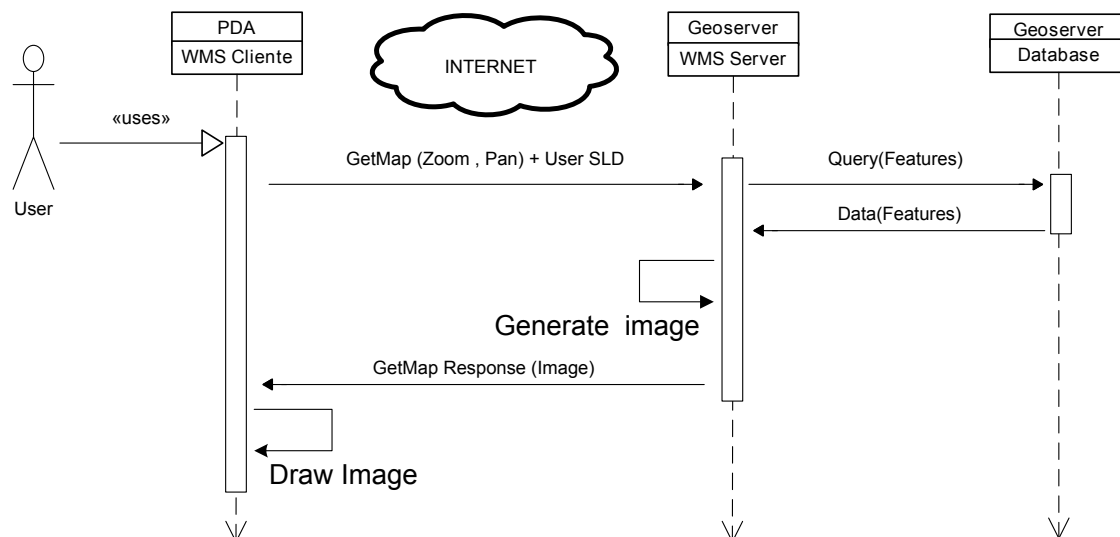


Figura 4.11: Fluxo de informação indicando operações de *zoom*.

descrevendo alguns cenários de utilização, de acordo com os objectivos propostos.

O protótipo foi desenvolvido na plataformas de desenvolvimento *Visual Studio 2008* através da linguagem *C#*, para a aplicação cliente a ser utilizada em dispositivos móveis e para o sistema operativo *Windows Mobile*. Por outro lado, foi desenvolvido um *Web Service* na plataforma *NetBeans*, utilizando a linguagem *java*. Este *Web Service* é responsável por gerar em tempo real o ficheiro *SLD* já referido.

#### 4.5.1 Ferramentas e Tecnologias Utilizadas

##### Servidor de Mapas *GeoServer*

Como já foi referido, o servidor de mapas utilizado no desenvolvimento deste projecto foi o *GeoServer*. Nesta secção será realizada uma descrição do modelo de arquitectura suportado pelo mesmo. Ainda é referido o modo como os dados foram configurados, visando compreender todo o trabalho que envolve a sua manipulação.

##### Arquitectura

O servidor *GeoServer* é constituído por um conjunto de módulos, cada um deles com uma função muito específica. A Figura 4.12 descreve as



componentes que definem a arquitectura do *GeoServer*. O módulo responsável pela gestão das outras componentes é denominado de *Spring*. A plataforma sobre a qual são desenvolvidas as aplicações neste contexto é o *GeoTools*. Existe também um módulo responsável pela contenção dos dados propriamente ditos. O conjunto de serviços normalizados *OGC WMS* e *WFS* constituem outro dos módulos do *GeoServer*. É disponibilizada uma interface de acesso para interacção com os utilizadores.

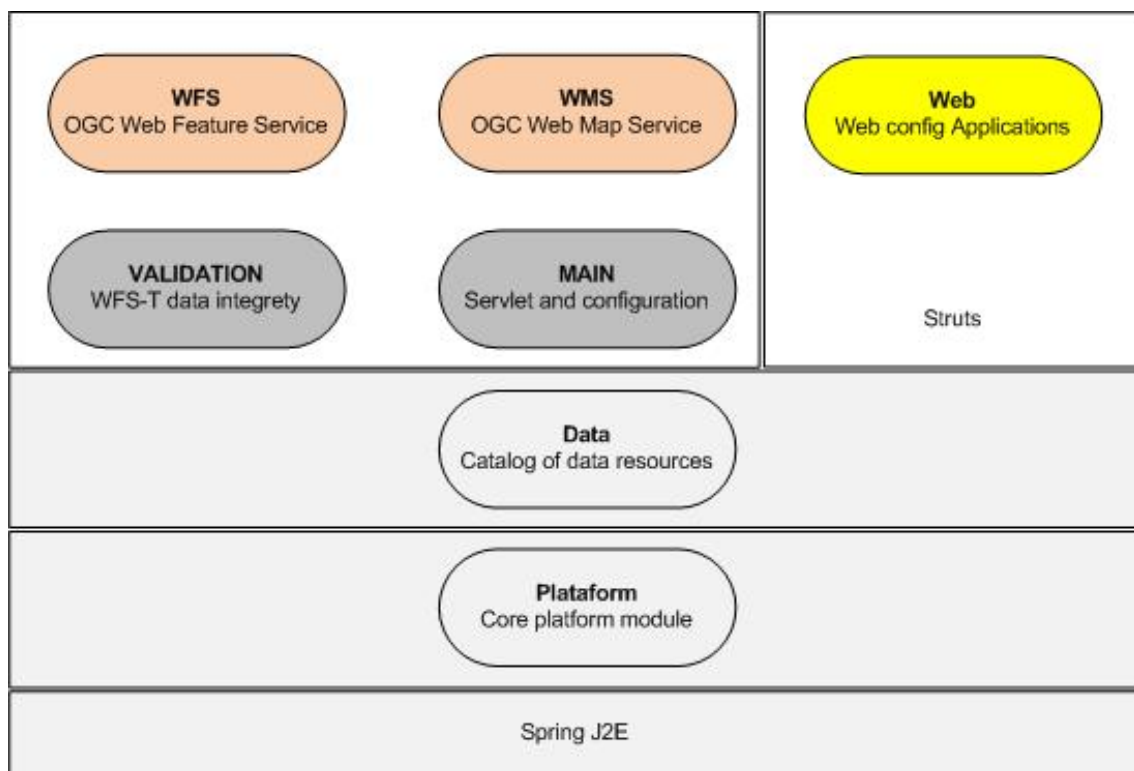


Figura 4.12: Arquitectura do servidor *Geoserver*.

## Configuração Dados

A interface gráfica de configuração do *GeoServer* é acessível através da *Web* e permitir configurar os serviços *WFS* e *WMS*.

A configuração dos dados envolve a activação dos serviços do *GeoServer*, através do comando *startup.sh*, localizado na directoria *bin* do servidor *Web Apache*. A partir deste momento é possível aceder ao seguinte endereço *Web* <http://192.168.65.135:8080/geoserver>. Este endereço e o porto foram os escolhidos para o acesso *Web* aos serviços, mas podem ser configurados de forma alternativa. Na Figura 4.13 pode

observar-se a página inicial do servidor. Para iniciar a configuração dos dados acede-se à opção *Config* e depois a opção *Data*.

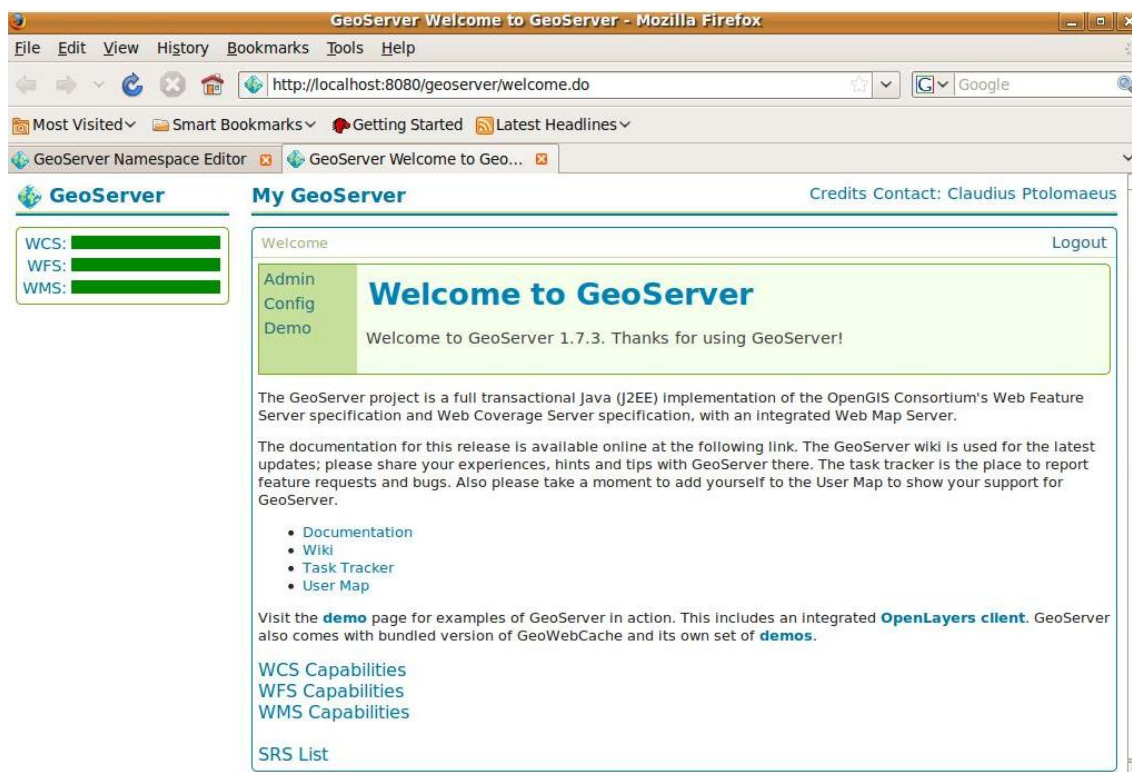
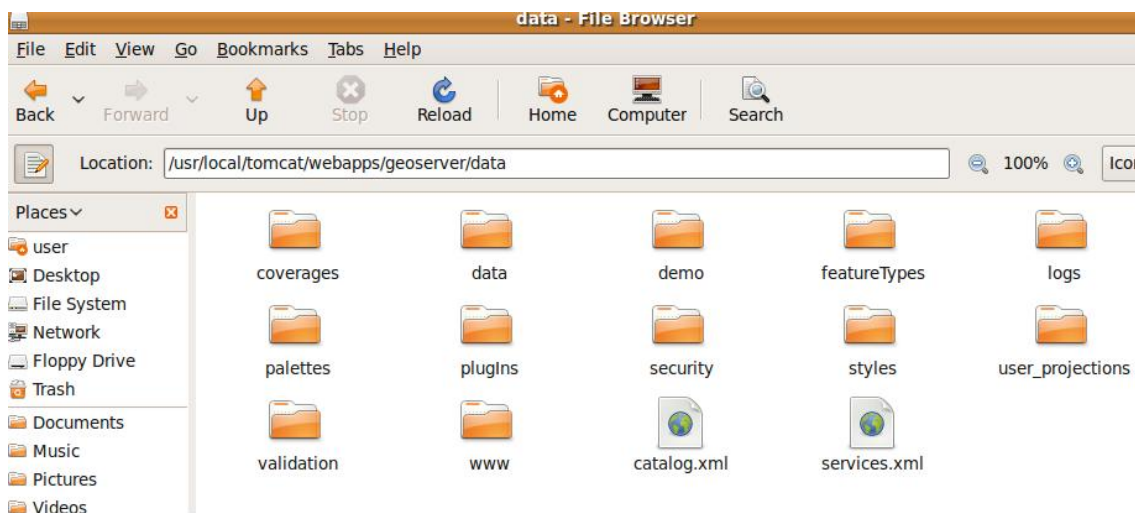


Figura 4.13: Página inicial do *GeoServer*.

Todas as configurações efectuadas traduzem-se em alterações nos dois principais ficheiros de configuração, o *catalog.xml* e o *service.xml*. O ficheiro *catalog.xml* contém a informação que é disponibilizada pelo *GeoServer*, nomeadamente, referências para os *shapefiles*, ligações às bases de dados *PostGis*. O ficheiro *service.xml* contém as opções de configuração do servidor, como por exemplo, o número máximo de elementos a retornar, níveis de acesso, administrador do sistema e respectiva palavra-passe. A Figura 4.14 exemplifica a estrutura de directorias do *GeoServer* onde se encontra a localização destes ficheiros.

Na configuração foram utilizadas duas estruturas de dados, para representar as camadas existentes. A camada referente aos pontos de interesse foi configurada através de uma ligação a uma base de dados *PostgreSQL-PostGis*, para representação de pontos geográficos. As restantes camadas foram configuradas directamente no servidor usando dados vectoriais no

Figura 4.14: Directorio de Dados do *GeoServer*.

formato *shapefile*. Os *shapefiles* poderiam ter sido convertidos em tabelas *PostGis* e adicionadas à base de dados que contém os pontos de interesse. Contudo, por opção não se procedeu a tal, uma vez que o servidor suporta a configuração das *shapefiles*. A camada de dados referente aos pontos de interesse foi mantida através de uma ligação a uma base de dados *PostgreSQL-PostGis*, porque estes dados foram obtidos de uma base de dados *MySQL* já existente no projecto *MoViSys*, e também porque não se possui esta informação tão completa em formato *shapefile*.

### *Namespace*

O espaço de nomes (*Namespace*) é um identificador que traduz o contexto de utilização dos dados geográficos. Não necessita de indicar um endereço específico na *Internet*, nem tem de ser válido, mas tem de ser definido. Associado à definição de espaço de nomes deve existir um prefixo, utilizado internamente pelo *GeoServer* para acesso aos dados, por exemplo: Portugal:estradas, Espanha:estradas. O prefixo é utilizado para acesso aos dados. Para editar o *Namespace*, selecciona-se a opção *new* no menu *Config/Data/Namespace* e selecciona-se o prefixo e o identificador, como demonstra a Figura 4.15.

### *DataStore*

O *DataStore* representa o espaço físico que contém os dados geográficos. Pode conter um ou mais tipos de dados, designados de *FeatureTypes* ou

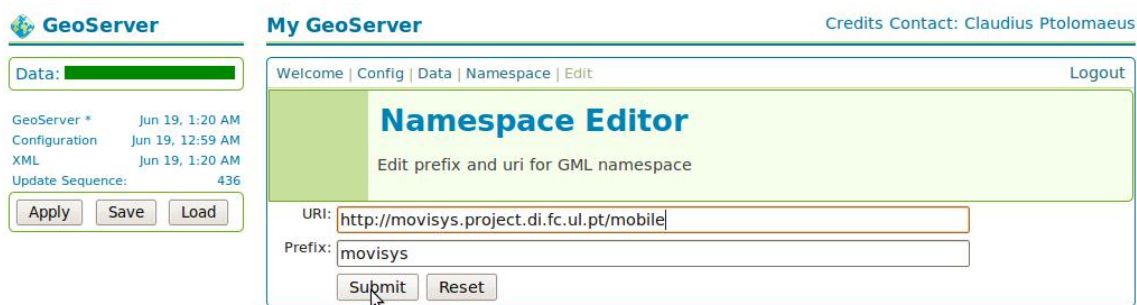


Figura 4.15: Configuração do espaço de nomes.

*layers*. Este tipo de dados pode ser uma ou mais tabelas, como numa base de dados, ou um único ficheiro no caso dos *shapefiles*. Este espaço físico é definido para que não seja necessário definir novos parâmetros de ligação para cada tabela numa base de dados. Cada tipo de dados está inserido em um espaço físico (*DataStore*) que define esses parâmetros. No caso dos *shapefiles*, cada ficheiro existente representa só um tipo de dados. Por exemplo, um *shapefile* com informação sobre Portugal, pode conter informação sobre as estradas, rios e pontos de interesse. Contudo, cada tipo de informação representa um tipo de dados, designado de *Feature-Type*.

Os dois tipos de *DataStore* utilizados são discutidos separadamente, pois cada um possui os seus parâmetros de configuração. Inicialmente é descrito a configuração dos *shapefiles* e depois é apresentada a configuração relativa à base de dados *PostgreSQL-PostGis*.

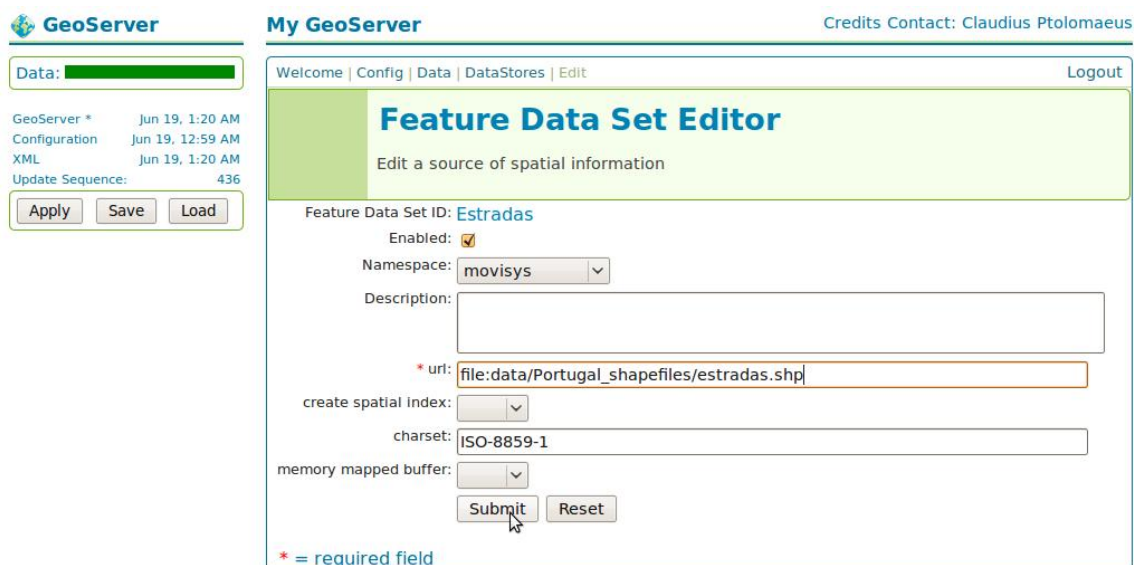
### *Shapefiles*

O principal factor de configuração de uma *shapefile* é a sua localização, indicado através de um *URL - Uniform Resource Locator*, que indica o endereço de um recurso. No *GeoServer* existe a possibilidade de copiar as *shapefiles* para a directoria *data*, ou indicar a sua localização no sistema de ficheiros. A primeira opção é mais flexível, pois permite que o servidor interprete a localização como um caminho relativo. Esse caminho é iniciado pelo nome *file:data/*, seguido do nome da pasta e o nome do ficheiro.

Para configurar selecciona-se no menu *DataStore* a opção *new*, e indica-se que tipo de tabela se pretende e qual o nome que identifica essa tabela, como se pode observar na Figura 4.16.

Figura 4.16: Configuração do espaço físico para uma *shapefile*.

Após configurar o tipo de tabela e o nome que a identifica, é necessário indicar a localização dos dados que a representa. A Figura 4.17 mostra o passo final para a configuração de uma tabela que representa uma *shapefile* referente a um tipo de dados. Todos os outros dados referentes às *shapefiles* são configurados do mesmo modo, mas diferem no nome e na localização.

Figura 4.17: Configuração da localização de uma *shapefile*.

## PostGis

Para a configuração dos dados referentes a esta base de dados, deve-se escolher a opção de modo similar ao da Figura 4.16. Contudo, a diferença baseia-se no tipo de tabela a seleccionar, *PostGis*, e nome que a identifica. Após seleccionar o botão *new* apresenta-se uma nova página que contém os parâmetros necessários à ligação do *GeoServer* com a base de dados referida. A Figura 4.18 mostra esses parâmetros.



Feature Data Set ID: pontos\_interesse

Enabled: ☒

Namespace: movisys

Description:

\* host: http://192.168.65.135

\* port: 5432

schema: public

\* database: dbmapas

\* user: user

passwd: ●●●●

max connections: 10

min connections: 4

validate connections: false

wkb enabled: true

loose bbox: true

estimated extent: false

Submit Reset

Figura 4.18: Configuração da ligação a uma base de dados *PostGis*.

Deve ser indicado o contexto de utilização dos dados seleccionando o espaço de nomes correspondente. Os restantes parâmetros referem-se ao endereço do servidor, ao porto de ligação, ao nome da base de dados, ao nome do utilizador e à palavra-passe correspondente. Após o clique no botão *submit* e depois *save* seleccionando a opção *apply*, termina a configuração de ligação a esta base de dados.

## Style

Cada camada de dados existentes no *GeoServer* representa um determinado tipo de dados e deve possuir pelo menos um estilo associado. O *GeoServer* possui um conjunto de estilos simples que pode ser utilizado para definir como as camadas são apresentadas e que informação relativa a estas camadas são visualizadas.

É possível a qualquer momento alterar o estilo de uma camada (*Layer*), editando a página correspondente a esta configuração. Na página inicial do servidor devese aceder-se ao menu *Config/Data/Styles*. O estilo definido é utilizado através de um ficheiro *SLD* que pode ser opcionalmente carregado ou editando directamente o *XML* correspondente no editor que surge na página de configuração. Caso se carregue um ficheiro *SLD* proveniente de uma localização diferente, o nome deste ficheiro não deve existir no servidor.

## FeatureType

Esta directoria contém informação sobre as camadas que existem no *GeoServer*. Para cada camada definida é criada uma sub-directoria, onde reside o ficheiro *info.xml*. Este ficheiro contém informação sobre a camada de dados correspondente. O tipo de informação existente, inclui: o sistema de referência ou projecção dos dados contidos na tabela, a área total que engloba os dados em questão, o estilo *SLD* utilizado pelo servidor para desenhar os dados, entre outros. Nas secções de configuração anteriores foi definido todo um conjunto de opções que vão ser utilizados nesta secção de configuração. Para iniciar a configuração, escolhe-se a tabela *DataStore*, que contém o tipo de dados *FeatureType* a configurar, como indica a Figura 4.19.

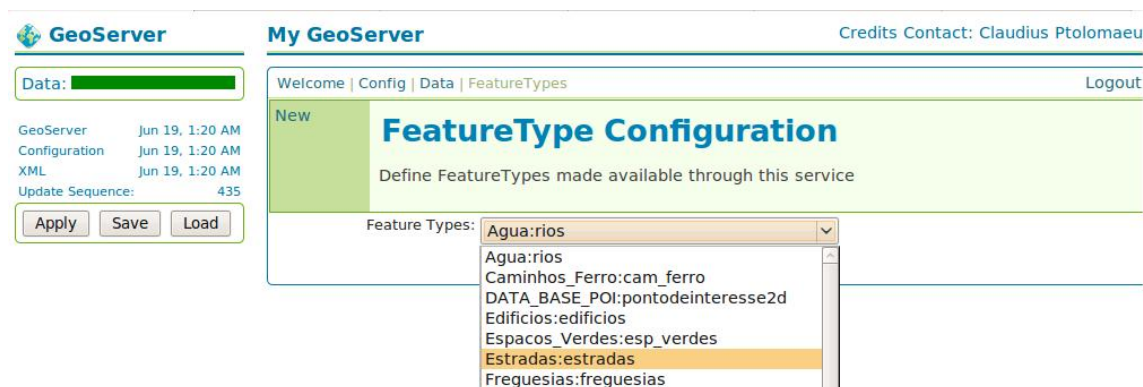


Figura 4.19: Selecção de uma camada para configuração.

Após esta configuração surge uma nova página onde é necessário definir as restantes características que definem a camada de dados. Deve ser escolhido o estilo associado, o sistema de referência, a área correspondente, o número máximo de dados a retornar, entre outros. As Figuras 4.20 e 4.21, mostram os modos de configuração para a escolha do estilo da camada, o sistema de coordenadas sobre qual o dados são projectados e a área que contém os dados.

Todas as camadas são projectadas sobre o mesmo sistema de referência. Assim deste modo os dados correspondentes aos pontos de interesse e as restantes camadas podem ser apresentados sobre um mesmo mapa sem que ocorra erros de localização. Quanto à área que indica a extensão dos dados, esta é gerada automaticamente, após o clique no botão *Generate*. Esta área já é apresentada em coordenadas do utilizador.

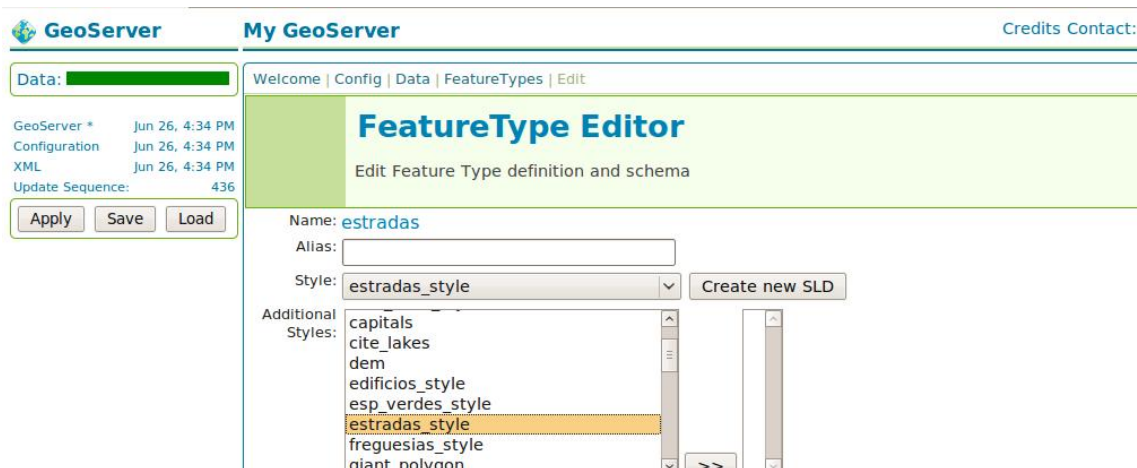


Figura 4.20: Configuração da informação de uma camada de dados.

SRS:  [Lookup SRS](#) [SRS Help](#) - [SRS List](#)

SRS WKT: PROJCS["Lisbon (Lisbon)/Portuguese National Grid", GEOGCS["Lisbon (Lisbon)", DATUM["Lisbon 1937 (Lisbon)", SPHEROID["International 1924", 6378388.0, 297.0, AUTHORITY["EPSG","7022"]], AUTHORITY["EPSG","6803"]], PRIMEM["Lisbon", -9.131906111111114, AUTHORITY["EPSG","6802"]], UNIT["degree", 0.017453292519943295], AXIS["Geodetic longitude", EAST], AXIS["Geodetic latitude", NORTH], AUTHORITY["EPSG","4803"]], PROJECTION["Transverse Mercator", AUTHORITY["EPSG","9807"]], PARAMETER["central\_meridian", 1.0], PARAMETER["latitude\_of\_origin", 39.666666666666664], PARAMETER["scale\_factor", 1.0], PARAMETER["false\_easting", 200000.0], PARAMETER["false\_northing", 300000.0], UNIT["m", 1.0], AXIS["Easting", EAST], AXIS["Northing", NORTH], AUTHORITY["EPSG","20790"]]

Native SRS: GEOGCS["GCS\_WGS\_1984", DATUM["D\_WGS\_1984", SPHEROID["WGS\_1984", 6378137.0, 298.257223563]], PRIMEM["Greenwich", 0.0], UNIT["degree", 0.017453292519943295], AXIS["Longitude", EAST], AXIS["Latitude", NORTH]]

SRS handling:

Title:

Bounding Box:

Data min X:	78398.04996593484	Data min Y:	3040.592306833074
Data max X:	369624.83459582145	Data max Y:	563975.377545206
Min Long:	-9.4977866	Min Lat:	36.9903992
Max Long:	-6.2266614	Max Lat:	42.0270255

Keywords:

Abstract:

Figura 4.21: Configuração da informação de uma camada de dados.

Os dados configurados referem-se a dados sobre Portugal Continental. A informação configurada refere-se a: Freguesias representando os contornos, Estradas, Ruas, Edifícios e Caminhos de Ferro. Estes foram os dados utilizados para teste e prova de conceito na manipulação das referências visuais sobre um mapa.

Para testar se a configuração de uma dada camada se realizou com sucesso e verificar como é apresentado a informação dessa camada no mapa, pode-se aceder à página inicial do *GeoServer* e escolher a opção *Demo*. Surge então uma página que contém todas as camadas configuradas no servidor e um conjunto de formatos que podem ser visualizadas.



### **Plataforma .NET Compact Framework**

Nesta secção são descritos os pormenores de implementação da aplicação desenvolvida para dispositivos móveis, assim como a plataforma de desenvolvimento utilizada.

O *Microsoft Visual Studio* é um ambiente integrado de desenvolvimento da *Microsoft* utilizado para desenvolver programas para consolas, interfaces gráficas, aplicações *Web*, *Web sites*, entre outros. É especialmente dedicado à plataforma .NET. Esta plataforma permite o uso de várias linguagens de programação, desde o *Visual Basic*, *C++*, *C#*, entre outros.

O .NET CF (*Compact Framework*) é uma versão da plataforma *Visual Studio .NET*, vocacionado para dispositivos móveis como *PDA*s e outros telefones móveis, utilizando o sistema operativo *Windows CE* (*Windows Embedded Compact*) ou *Windows Mobile*. As bibliotecas existentes na *Compact Framework* (CF) são menores do que as existentes na versão .NET. Face às características destes dispositivos é necessário reduzir as bibliotecas quando comparadas com a versão normal.

A escolha desta plataforma para construção da aplicação cliente baseou-se no facto de existir uma familiarização com a mesma e do anterior projecto ter sido desenvolvido nesta plataforma.

Durante o desenvolvimento das componentes da aplicação cliente, teve de se ter em conta a relação que as mesmas devem ter no contexto móvel. O uso dessas componentes envolve pessoas, acessos à *Internet*, uso de tecnologias *standard*, os dados propriamente ditos e os dispositivos.

De acordo com as tecnologias já analisadas, foi implementado o acesso aos serviços *WMS da OGC*. Foi então construída uma classe que define os serviços *GetCapabilities*, *GetMap* e *GetFeature*. Denominou-se essa classe de *Wms\_Services*. Para manipular os parâmetros dos pedidos e facilitar ao cliente métodos de acesso e obtenção da informação, foi desenvolvido uma classe cliente de nome *Wms\_Cliente*. Este conjunto de parâmetros é passado ao *Wms\_Services* para realizar os pedidos dos mapas e visualizar a informação.

É na classe *Wms\_Cliente* que são geradas as operações de aproximação e afastamento, movimento do mapa, adição ou remoção de camadas de dados (*Layers*) e selecção das propriedades das camadas que se pretendem visualizar. Vários métodos auxiliares foram criados de modo a facilitar o uso destas operações. As operações por se basearem em tecnologias *Web Service*, facilitam o processo de acesso à informação, através de pedidos *HTTP*.

A classe de nome *Layer* foi construída e representa uma camada de dados com os seus atributos e métodos de acesso. A classe *Wms\_Cliente* possui um conjunto destas *Layers*, obtidas inicialmente através do ficheiro *XML* requisitado pela operação *GetCapabilities*.

De modo a facilitar as opções que permite ao utilizador seleccionar a informação a visualizar foram construídas classes auxiliares que não constam no modelo de domínio. Depois de implementadas chegou-se à conclusão que essas classes auxiliares deveriam representar atributos e métodos de acesso da classe *Layer*. De facto e apesar de não serem sub-classes, representam de certo modo uma generalização da classe *Layer*. Essas classes modelam em termos de objectos as propriedades das camadas.

A classe que representa uma transformação janela enquadramento, permite transformar as coordenadas do utilizador em coordenadas do servidor, para realizar o pedido da área do mapa a visualizar. Quando o utilizador interage com o mapa, realizando opções de afastamento, aproximação e movimento do mapa, é calculada uma nova área de visualização do mapa no dispositivo móvel, baseado no deslocamento realizado. O resultado é utilizado para obter a área correspondente no servidor e é passado com parâmetro no pedido de um novo mapa.

Por motivos de simplificação, muitos atributos da classe cliente foram omitidos, sendo apenas apresentados os mais importantes. A Figura 4.22 representa o modelo de classes utilizado no desenvolvimento da aplicação cliente. Os parâmetros referidos são os descritos na secção três. O serviço *GetMap* da classe *Wms\_Services* recebe esses parâmetros e constrói o pedido. Através de uma ligação *HTTP*, o pedido é enviado para o servidor de mapas. No caso do serviço que gera o ficheiro *SLD*, este só é activado

caso o utilizador seleccione relativamente a uma ou mais camadas, quais as propriedades que pretende visualizar. Caso o utilizador seleccione uma camada e não seleccione que propriedades quer visualizar, é apresentada toda a informação dessa camada, dependendo do nível de zoom apresentado.

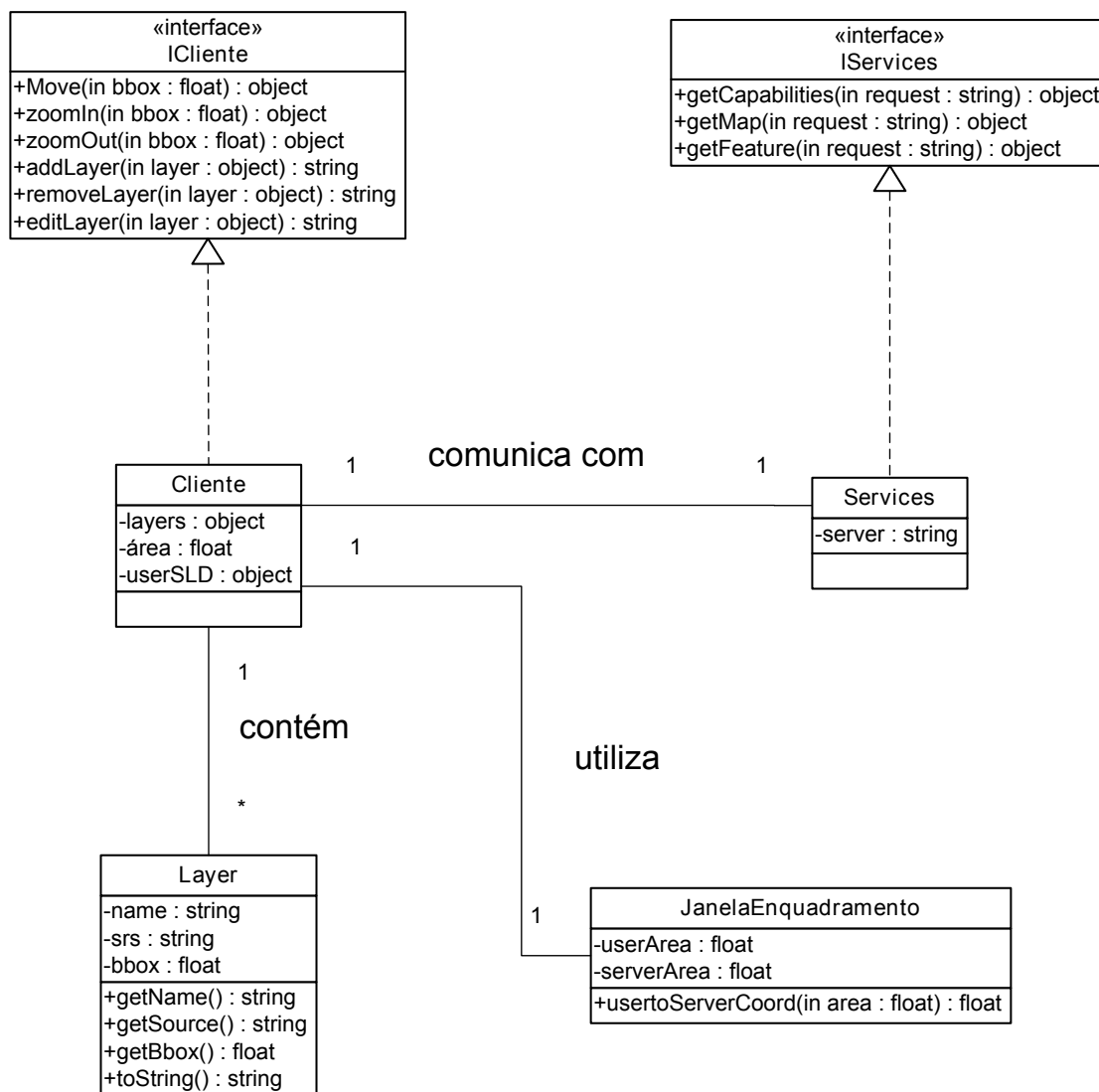


Figura 4.22: Diagrama de classes da aplicação cliente

## Plataforma *NetBeans*

Nesta secção realiza-se uma introdução sobre a ferramenta *NetBeans*, e descreve-se como foi utilizada na construção do serviço responsável por gerar o ficheiro *SLD*. Descreve-se também mais pormenorizadamente o funcionamento da linguagem *SLD*.

O *NetBeans* é uma ferramenta de desenvolvimento que permite construir programas para computadores de secretária, aplicações *Web* e dispositivos móveis. Foi construída utilizando a linguagem *Java* e é de domínio público. Apesar de escrito em *Java*, possui suporte a várias outras linguagens de programação, tais como: *C*, *C++*, *Ruby*, *PHP*, *XML* e *HTML*.

O *NetBeans* fornece uma base sólida para a construção de projectos, programas ou módulos de programação, pois possui um grande conjunto de bibliotecas, as (*API's*), e uma vasta documentação bem estruturada. Contudo, é necessário uma máquina virtual *Java* (*JVM*) para funcionar.

Esta ferramenta foi escolhida com o intuito de construir um serviço *Web*, capaz de gerar, de acordo com as preferências do utilizador, um ficheiro *XML* correspondente ao *SLD*. Este ficheiro permite indicar ao servidor de mapas quais são os dados que são desenhados e de que modo estes são desenhados no mapa. A escolha desta ferramenta foi opcional e o uso da linguagem *Java* para a construção de um serviço *Web* deve-se à familiarização com a mesma. A utilização desta ferramenta para a construção do serviço foi bastante facilitada pela quantidade de exemplos existentes na *Web*, tais como em [29].

Do mesmo modo que foi utilizado o servidor de mapas, esta ferramenta permite gerar a partir do código desenvolvido um ficheiro com extensão *.WAR*. Este é adicionado ao servidor *Web* para que o serviço fique disponível. O Serviço é acedido pelo cliente através de uma ligação *Web*, que procura através do endereço do serviço, os métodos que este fornece. Através desta ligação são enviados como parâmetros dois ficheiros *XML* contendo a informação desejada e que foi manipulada previamente do lado da aplicação. Com base na informação recebida é construído um ficheiro *XML*. Esse ficheiro pode conter informação sobre as camadas, tais como, pontos de interesse, estradas, edifícios.

Para o acesso a este serviço, a classe cliente disponibiliza ao utilizador uma lista de camadas disponíveis. O utilizador selecciona as camadas desejadas e sobre estas pode seleccionar as propriedades que pretende visualizar. É ainda possível escolher o modo como as propriedades são apresentadas no mapa, indicando as cores do objecto, a sua dimensão e se pretende ou não representação textual. O sistema constrói sobre essa informação uma lista de objectos que é enviado ao serviço requisitado.

Esta lista é enviada em formato *XML*, garantindo que os dados possam ser enviados pela rede e interpretados por quem as recebe. O serviço ao receber estes parâmetros retira a informação necessária de modo a construir dois objectos. Um dos objectos refere-se aos pontos de interesse, o outro às restantes camadas. Esta distinção deve-se ao facto de os parâmetros manipulados pelos pontos de interesse serem distintos das outras camadas, apesar de representar também uma camada de dados. Os métodos desses objectos permitem construir o ficheiro *SLD*. A localização do ficheiro é enviado ao cliente que por sua vez realiza o pedido do mapa.

A Figura 4.23 traduz a sequência de operações realizada entre as componentes, para requisitar este serviço e gerar o ficheiro. O ficheiro *SLD* quando é recebido pelo servidor de mapas possui preferência sobre as definições de configuração existentes. O servidor interpreta este ficheiro e realiza o desenho da informação. O resultado é uma imagem no formato escolhida pela aplicação cliente.

Um ficheiro *SLD* contém um conjunto de nomes de camadas denominadas de *NamedLayer* e estas possuem um conjunto de regras que definem o que deve ser apresentado.

Para a construção deste serviço foi necessário compreender e analisar o esquema que especifica o modo de construção dos ficheiros *SLD*. O elemento *StyledLayerDescriptor*, indica a raiz da hierarquia que constitui o ficheiro *SLD*, e contém o espaço de nomes. A raiz pode possuir um conjunto de camadas definidas pelo elemento *NamedLayer*, para o qual se deve indicar o nome. Este nome deve ser exactamente o nome da camada contendo o seu prefixo. Opcionalmente, pode-se definir o nome, o título e uma definição *abstract*. Contudo, deve existir pelo menos um *NamedLayer*.

Dentro da hierarquia do *NamedLayer* define-se qual o estilo a aplicar, indicado pelo elemento *UserStyle*, ao qual se pode também definir opcionalmente o nome, o título e uma definição *abstract*. Este último contém obrigatoriamente um elemento denominado de *FeatureTypeStyle*, no qual se definem as várias regras que indicam quais as propriedades das camadas a apresentar. Cada regra contém um filtro (*Filter*) que indica qual a propriedade a aplicar à regra e um *Symbolizer* indicando como a geometria do objecto é desenhada.

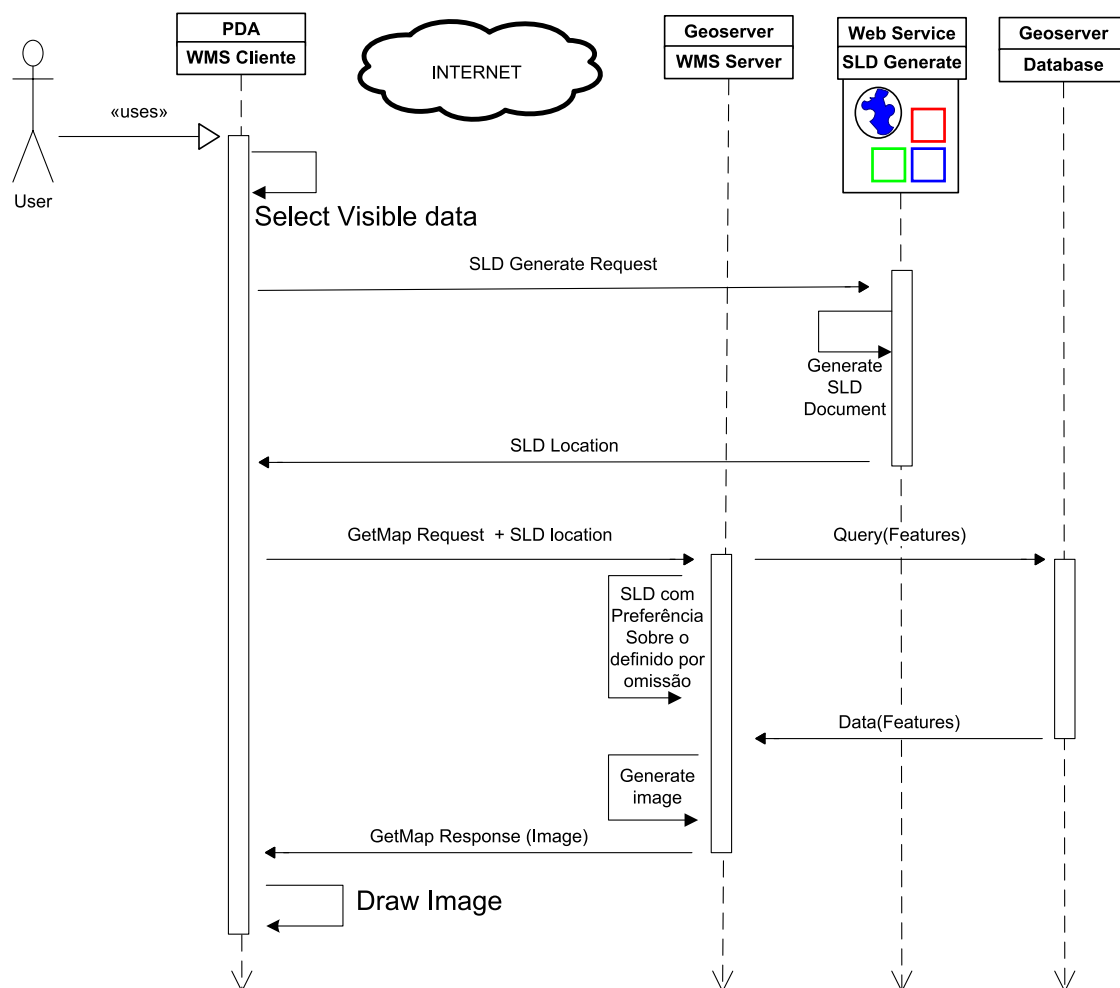


Figura 4.23: Diagrama de sequência representando a interacção entre as componentes do sistema, no que se refere à geração e interpretação do ficheiro *SLD*.

O filtro é opcional e indica qual a informação seleccionada. Quando o filtro não é aplicado, a regra aplica-se a todas as propriedades da camada. Pode ainda ser definido combinando operadores lógicos e de comparação, entre outros.

O *Symbolizer* descreve o modo de apresentação da geometria, caso se trate de uma linha (*LineSymbolizer*), de um ponto (*PointSimbolizer*) ou um polígono (*PoligonSymbolizer*). É possível definir o modo de apresentação da geometria, as cores, dimensão do objecto e representação textual, ícones para os objectos, entre outros.

Os outros elementos contidos sobre a hierarquia das regras, tais como, o *MinScaleDenominator* e o *MaxScaleDenominator*, permitem definir os

níveis de *zoom* sobre o qual as regras são apresentadas. A informação sobre uma camada pode ser definida por várias regras. Todas as regras devem ser construídas de acordo com a hierarquia que define o esquema da estrutura de um ficheiro *SLD* tal como mostra a Figura 4.24.

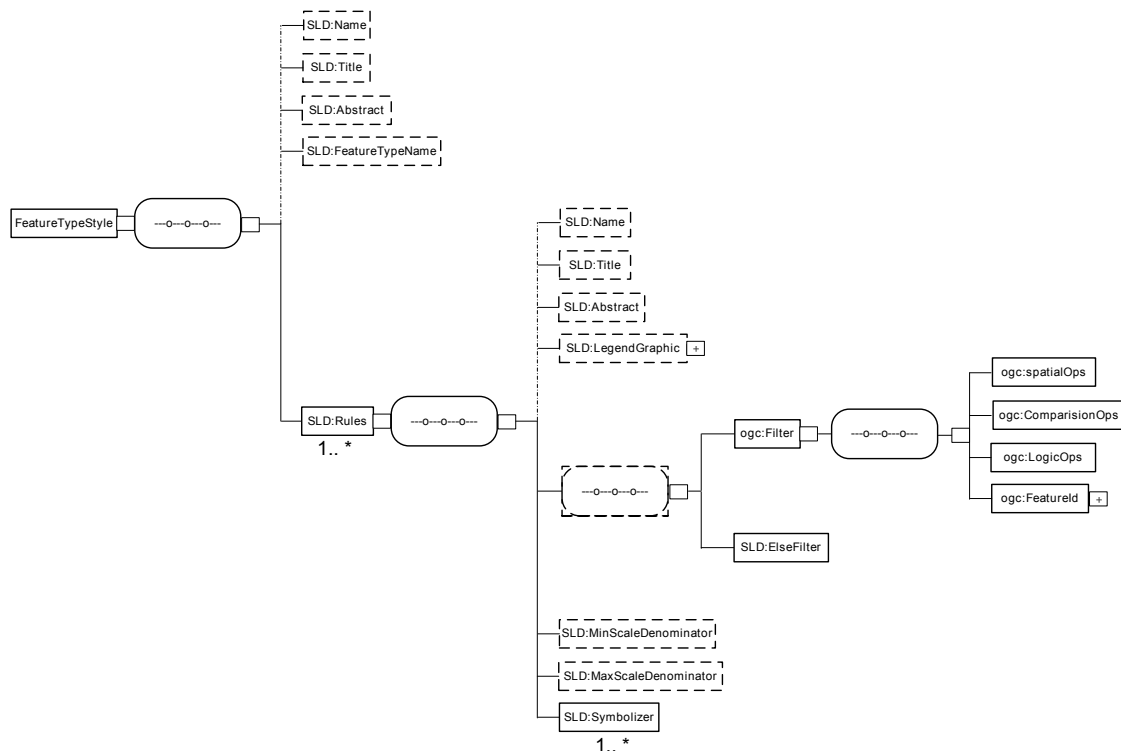


Figura 4.24: Esquema hierárquico das regras do ficheiro *SLD*.

Para exemplificar a construção de um ficheiro *SLD*, apresenta-se um excerto do mesmo, definido pelos elementos descritos anteriormente. É de referir ainda que este ficheiro pode ser modificado sempre que o utilizador seleccione outras camadas e novas propriedades, pois o conteúdo anterior é substituído. De modo a simplificar o acesso a este documento, quando o mesmo é gerado, é enviado para uma pasta localizada no servidor de mapas, com o nome do dispositivo do utilizador, definido pelo seu endereço *MAC*. Deste modo é possível a vários utilizadores em simultâneo editar as suas próprias preferências sem interferir com as dos outros utilizadores. A Figura 4.25 traduz esse excerto.

```

<?xml version="1.0" encoding="UTF-8"?>
<StyledLayerDescriptor version="1.0.0"
  xsi:schemaLocation="http://www.opengis.net/sld
StyledLayerDescriptor.xsd"
  xmlns="http://www.opengis.net/sld"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<NamedLayer>
  <Name>movisys:Estradas</Name>
  <Title>Camada das Estradas</Title>
  <UserStyle>
    <Name>User Style</Name>
    <Title>The style</Title>

    <FeatureTypeStyle>
      <Rule>

        <ogc:Filter>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>Type</ogc:PropertyName>
            <ogc:Literal>highway</ogc:Literal>
          </ogc:PropertyIsEqualTo>
        </ogc:Filter>

        <LineSymbolizer>
          <Stroke>
            <CssParameter name="stroke">
              <ogc:Literal>#000000</ogc:Literal>
            </CssParameter>
            <CssParameter name="stroke-width">
              <ogc:Literal>1.0</ogc:Literal>
            </CssParameter>
          </Stroke>
          ...
        </LineSymbolizer>

      </Rule>
    </FeatureTypeStyle>
  </UserStyle>
</NamedLayer>

<NamedLayer>
  <Name>movisys:PontoInteresse</Name>
  ...

```

Figura 4.25: Exemplo de um ficheiro *SLD*.



### 4.5.2 Protótipo

Nesta secção apresentamos alguns cenários de interacção do utilizador com o protótipo que cobrem as diversas funcionalidades apresentadas na secção anterior.

Ao iniciar a aplicação é apresentada ao utilizador a interface inicial. O utilizador deve iniciar a ligação com o servidor de mapas. O endereço do servidor foi introduzido directamente no código, contudo poderia ser o utilizador a indicar este servidor. Após estabelecer a ligação, a aplicação carrega para o dispositivo a informação sobre as camadas de dados que o utilizador pode seleccionar e manipular, como mostra a Figura 4.26. O utilizador deverá escolher a opção *startMap* de modo a iniciar a interface de visualização do mapa, como mostra a Figura 4.27. A imagem inicialmente apresentada refere-se ao mapa de Portugal continental e contém todas as camadas de dados existentes.

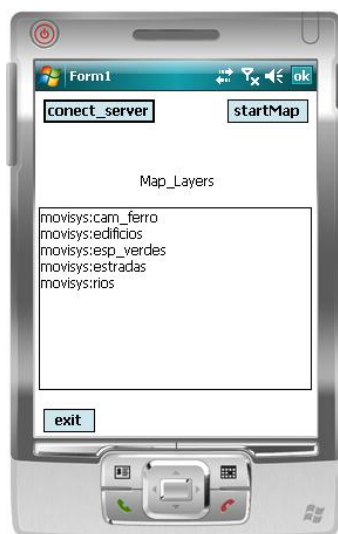


Figura 4.26: Interface inicial.



Figura 4.27: Visualização do mapa.

### Operações de afastamento/aproximação

A informação geográfica visualizada num determinado mapa, depende do nível de aproximação e de afastamento. Esta informação foi configurada no servidor de mapas e é a informação a visualizar por omissão. A ordem pelo qual as camadas são apresentadas é determinada pela ordem que

são seleccionadas. Para que o utilizador não se preocupe com a ordem de selecção das camadas, foi atribuída a estas uma ordenação. Assim, no caso de existir intersecção das camadas nenhuma delas cobrirá na totalidade a outra. Por exemplo, a camada referente aos espaços verdes deve surgir primeiro que a camada dos pontos de interesse. Uma vez que os espaços verdes geometricamente representam um polígono com preenchimento, esta camada poderia tapar a informação referente aos pontos de interesse.

O utilizador interage com a aplicação realizando as operações de afastamento, aproximação, através dos botões que se encontram no canto superior direito do mapa. O utilizador pode ainda movimentar-se no mapa arrastando o mesmo para outras áreas. É possível também adicionar ou filtrar informação através das opções fornecidas pelo botão *select*.

### Seleccção de camadas

O objectivo da selecção de camadas de informação é permitir que o utilizador controle a informação geográfica visualizada. Considere-se a situação em que um utilizador se encontra a explorar o mapa, num determinado nível de aproximação, como mostra a Figura 4.28 e é-lhe apresentada a seguinte informação:

- Uma camada de dados referente às estradas, que possui como propriedades: ruas (delineadas a branco), uma avenida (delineada a rosa) e uma estrada municipal sem informação (delineada a cinzento);
- Uma camada de dados referente a espaços verdes;
- Uma camada de dados referente aos edifícios com cor de preenchimento a azul.

Caso o utilizador pretenda visualizar só os espaços verdes e edifícios desta localização no mapa, pode omitir a camada de informação referente a estradas. Para isso basta seleccionar o botão *select* na interface. Na interface de selecção (*select*), o utilizador pode escolher diversas opções, tais como: visualizar pontos de interesse, seleccionar as camadas de informação que pretende visualizar ou editar as propriedades das camadas, como mostra a Figura 4.29. Neste caso, o utilizador deve seleccionar a opção *select layers* e escolher as camadas pretendidas, como mostra a Figura 4.30.



Figura 4.28: Visualização do mapa num dado nível de aproximação.

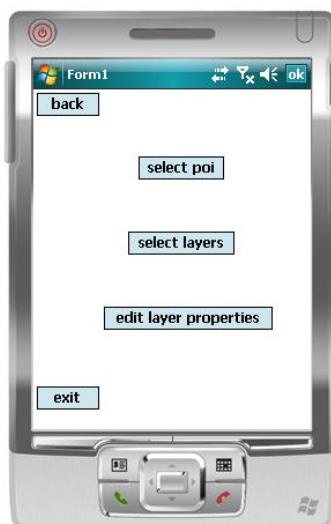


Figura 4.29: Interface de selecção de opções.

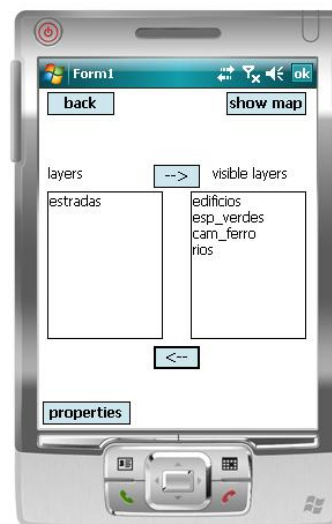


Figura 4.30: Interface de selecção de camadas.

Após esta operação deverá escolher a opção *show map* e ser-lhe-á apresentado um mapa apenas com a informação seleccionada, como mostra a Figura 4.31.

## Visualização de Pontos Interesse

Os pontos de interesse tal com referido anteriormente são representados numa camada de dados, contudo a sua selecção é realizada à parte, pois possui os seus próprios atributos.



Figura 4.31: Interface obtida após a omissão da camada de estradas.

Consideremos o seguinte cenário: Um utilizador está a explorar uma área do mapa como mostra a imagem da Figura 4.32, e pretende saber se existe algum hipermercado, e ainda saber quais os postos de combustível da *Galp* existentes que possuem cartão de desconto.



Figura 4.32: Visualização do mapa num dado nível de aproximação.

Neste cenário o utilizador selecciona o botão *options* e é-lhe mostrado o ecrã da Figura 4.29. Como o utilizador pretende obter pontos de interesse selecciona o botão *select poi*. É-lhe apresentado um novo ecrã onde pode seleccionar consoante os atributos, os pontos de interesse que pretende visualizar. Adicionalmente pode definir o tamanho dos ícones que

representam os pontos de interesse. As Figuras 4.33 e 4.34 indicam as escolhas efectuadas para este cenário.



Figura 4.33: Interface de selecção de pontos de interesse.



Figura 4.34: Interface de selecção de pontos de interesse.

Após seleccionar os pontos de interesse o utilizador deve guardar as alterações para que estas tenham efeito, através do botão *save changes*. Ao clicar no botão *show map* são apresentados os resultados, como mostra a Figura 4.35. A figura indica a presença de um hipermercado nas redondezas e a existência de 2 postos de combustível *Galp*. O utilizador pode a qualquer momento seleccionar outros pontos de interesse, seleccionando novamente esses pontos e guardar as alterações. Caso o utilizador não pretenda visualizar nenhum ponto de interesse, basta a qualquer momento retirar de selecção as *checkboxes* correspondentes a esses pontos e guardar as alterações.

### Seleccção das propriedades de uma camada de dados

Um outro cenário é a possibilidade do utilizador seleccionar apenas determinada propriedade de uma camada de dados, num determinado nível de aproximação. A Figura 4.36 mostra um mapa com três camadas de dados, estradas, espaços verdes e caminhos de ferro. A camada das estradas mostra informação sobre vias rápidas, demarcadas a verde, ruas demarcadas a branco, avenidas demarcadas a cor de rosa e sem informação textual e a cinzento ruas principais.



Figura 4.35: Interface de visualização de pontos de interesse no mapa.



Figura 4.36: Visualização do mapa num dado nível de aproximação.

O utilizador pode, por exemplo, querer visualizar somente as ruas nesta área do mapa, retirando da área de visualização a restante informação sobre a camada das estradas. Para isso selecciona o botão das propriedades e é-lhe mostrado o ecrã de selecção como no exemplo da Figura 4.29. Como o utilizador pretende editar as propriedades de uma dada camada selecciona o botão correspondente. É-lhe apresentado um novo ecrã que lhe permite seleccionar qual a camada que pretende editar, como mostra a Figura 4.37. O utilizador ao seleccionar a camada das estradas pretendida, pode filtrar as propriedades dessa camada, como se pode ver na imagem da Figura 4.38.

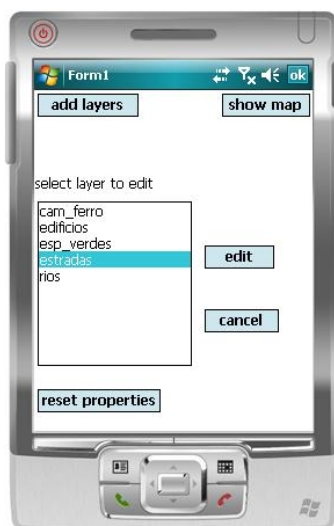


Figura 4.37: Interface de selecção das camadas a editar.

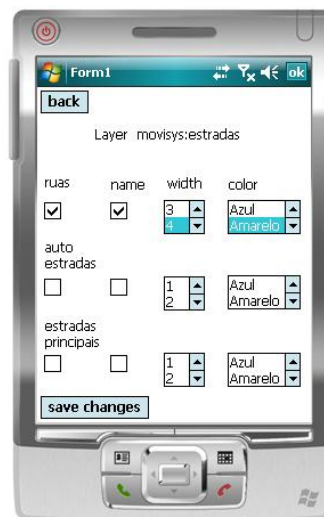


Figura 4.38: Interface de selecção das propriedades de uma camada.

As opções apresentadas permitem seleccionar a informação a visualizar, ruas, estradas principais (avenidas e estradas municipais) e auto estradas, incluindo as vias rápidas e itinerários. O utilizador selecciona então somente as ruas e escolhe ainda se pretende o nome das mesmas, qual a largura que pretende para as linhas que as representam e ainda qual a cor pretendida. Ao guardar as alterações é lhe apresentado o ecrã anterior. Ao seleccionar a opção *show map*, é apresentado no mapa a imagem correspondente à selecção, como mostra a Figura 4.39.

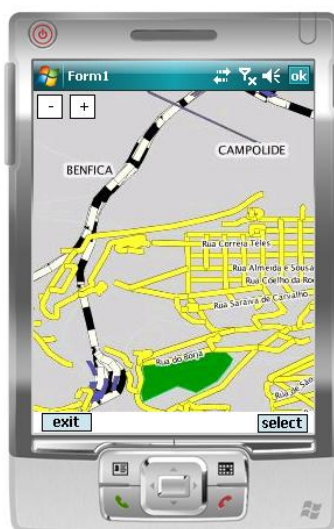


Figura 4.39: Visualização do mapa num dado nível de aproximação.

Caso o utilizador pretenda voltar às definições do sistema basta a qualquer momento seleccionar a opção *reset properties* do ecrã da Figura 4.37.

## Sumário

Os exemplos dos cenários de utilização apresentados são bastante simples, mas foram elaborados com o objectivo de exemplificar as diversas funcionalidades disponíveis. Dependendo do nível de afastamento/aproximação a que o utilizador se encontra no mapa, estas opções permitem obter imagens mais simples de interpretar.

Relativamente à camada das estradas que constituiu a maior parte dos exemplos mencionados, é de referir que a diferença entre as cores das linhas apresentadas, sua espessura e a informação textual, foi determinada consoante os níveis de aproximação/afastamento a que o mapa se encontrava. Contudo, a principal razão deve-se ao facto desta camada possuir um conjunto de propriedades que diferencia o tipo de estradas que existe. Esta camada possui estradas que se referem a auto\_estradas (realizam a ligação entre cidades, A1, A15) e possui vias rápidas que não são consideradas auto estradas e que realizam as ligações dentro das cidades, (itinerários principais e complementares, IC19, IP7). Adicionalmente, possui estradas principais, como avenidas e ainda as ruas que indicam normalmente as zonas residenciais.



# Capítulo 5

## Conclusões e trabalho futuro

A evolução dos dispositivos móveis tem vindo a proporcionar novas oportunidades nos domínios onde a informação geográfica desempenha um papel importante. As limitações apresentadas por estes dispositivos quando comparadas com os computadores de secretária, sugerem que se adoptem técnicas de visualização eficientes, que permitam aos utilizadores aceder e compreender a informação necessária para a realização das suas tarefas. Desta forma é fundamental que os utilizadores tenham a possibilidade de manipular as referências visuais existentes sobre um mapa, nomeadamente, especificar qual a informação geográfica visualizada a cada momento e de acordo com o contexto de utilização.

### 5.1 Conclusão

Este trabalho, enquadrado num projecto de investigação contribuiu para o desenvolvimento de um protótipo que permite aos utilizadores filtrar a informação geográfica que pretendem visualizar a cada momento. Podemos concluir que os objectivos propostos nesta dissertação foram atingidos. De acordo com o protótipo desenvolvido, é possível manipular as referências visuais existentes num mapa. O protótipo baseou-se em 3 componentes principais:

- Uma aplicação cliente que permite aos utilizadores seleccionar qual a informação a visualizar a cada momento;
- Um servidor de mapas sobre o qual se realizam pedidos baseados em parâmetros específicos e que são manipulados pela aplicação cliente;
- Um serviço capaz de gerar em tempo de execução um ficheiro que contém os filtros definidos pelo utilizador, indicando o modo como a informação deve ser apresentada num mapa.

A documentação existente sobre os servidores de mapas e os serviços normalizados *WMS/WFS*, permite uma maior compreensão do modo de acesso e configuração da informação geográfica a ser utilizada.

Uma das grandes dificuldades quando se desenvolvem aplicações que utilizam estas tecnologias, é decidir qual a informação geográfica a apresentar ao utilizador a cada momento, de acordo com as configurações por omissão. É necessário um estudo prévio e um grande conjunto de testes para obter bons resultados. Além disso é necessário conhecer toda a informação disponível e que vai ser apresentada ao utilizador, de acordo com as suas necessidades de utilização, uma vez que é necessário saber qual a informação que se pode permitir aos utilizadores manipular.

Outra das dificuldades que levou a algum atraso no decorrer deste projecto foi a falta de disponibilidade dos dados geográficos, os *shapefiles*, para teste e configuração no servidor.

## 5.2 Trabalho futuro

Toda a informação manipulada e disponibilizada aos utilizadores, pode ser optimizada, relativamente à localização da informação textual, localização dos ícones, afastamento dos objectos, melhorando os aspectos visuais da informação no mapa.

Uma tarefa para trabalho futuro é a integração deste projecto com o desenvolvido anteriormente, no que se refere às técnicas de permitem reduzir a ocorrência de sobreposição dos ícones sobre o mapa, com a capacidade de filtrar e seleccionar parte da informação geográfica visualizada nos mapas. Deste modo o utilizador teria maior controlo sobre a análise das tarefas que pretende realizar.

Outro aspecto que merece um trabalho mais detalhado é a interface do protótipo. De facto e uma vez que apenas se pretende uma prova de conceito, este não foi um aspecto explorado neste projecto.

### 5.3 Análise pessoal

Este projecto de investigação foi bastante interessante, não só pelo desafio proposto como também pela contribuição sobre os conhecimentos obtidos. A existência das reuniões periódicas facilitou a compreensão e desenvolvimento dos objectivos propostos, levando à orientação na direcção correcta.

O atraso inicial deste projecto, teve como motivo a linha de estudo proposta inicialmente de explorar o formato vectorial *SVG*. Contudo, este estudo revelou uma elevada complexidade da sua utilização para dispositivos móveis, e permitiu seguir uma outra via mais simples de concretização.



# Abreviaturas

<b>GIF</b>	Graphics Interchange Format
<b>HTML</b>	HyperText Markup Language
<b>JPEG</b>	Joint Pictures Expert Group
<b>MAC</b>	Media Access Control
<b>OGC</b>	Open Geospatial Consortium
<b>PNG</b>	Portable Network Graphics
<b>SIG</b>	Sistema de Informação Geográfica
<b>SLD</b>	Styled Layer Descriptor
<b>SOAP</b>	Simple Object Access Protocol
<b>SVG</b>	Scalable Vector Graphics
<b>URL</b>	Uniform Resource Locators
<b>W3C</b>	World Wide Web Consortium
<b>WFS</b>	Web Feature Service
<b>WMS</b>	Web Map Service
<b>WSDL</b>	Web Services Description Language
<b>XML</b>	eXtensible Markup Language



# Bibliografia

- [1] Bases de dados geográficas. [http://en.wikipedia/wiki/Spatial\\_database](http://en.wikipedia/wiki/Spatial_database). Data último acesso: 05 Junho, 2009.
- [2] Geóide. <http://pt.wikipedia.org/wiki/Geoide>. Data último acesso: 10 Maio, 2009.
- [3] Geodesia. <http://pt.wikipedia.org/wiki/Geodesia>. Data último acesso: 18 Maio, 2009.
- [4] Geography markup language. [http://en.wikipedia.org/wiki/Geography\\_Markup\\_Language](http://en.wikipedia.org/wiki/Geography_Markup_Language). Data último acesso: 4 Novembro, 2008.
- [5] Geoserver. <http://geoserver.org/display/GEOS/Welcome>. Data último acesso: 14 Maio, 2009.
- [6] Geoserver configuração dados. <http://geoserver.org/display/GEOSDOC/2.1+Data+Configuration>. Data último acesso: 16 Maio, 2009.
- [7] Geoserver download. <http://docs.geoserver.org/1.7.x/user/installation/geoserver-install.html>. Data último acesso: 11 Fevereiro, 2009.
- [8] Geovis. <http://xldb.fc.ul.pt/wiki/GeoVis>. Data último acesso: 14 Novembro, 2008.
- [9] Gimodig architecture specification. <http://gimodig.fgi.fi/deliverables>. Data último acesso: 28 Outubro, 2008.
- [10] Html. <http://en.wikipedia.org/wiki/HTML>. Data último acesso: 4 Novembro, 2008.
- [11] Imagens projecção cartográfica. <http://www.professores.uff.br/cristiane/Estudodirigido/Cartografia.htm>. Data último acesso: 14 Maio, 2009.

- [12] Informação. <http://pt.wikipedia.org/wiki/Informacao>. Data último acesso: 20 Maio, 2009.
- [13] Levantamento topográfico. [http://pt.wikipedia.org/wiki/Levantamento\\_topografico](http://pt.wikipedia.org/wiki/Levantamento_topografico). Data último acesso: 18 Maio, 2009.
- [14] Mapserver. <http://mapserver.org/>. Data último acesso: 16 Maio, 2009.
- [15] Meridiano. <http://pt.wikipedia.org/wiki/Meridiano>. Data último acesso: 22 Maio, 2009.
- [16] Opensls presentation service. <http://www.opengeospatial.org/standards/ols>. Data último acesso: 5 Novembro, 2008.
- [17] Postgis. [postgis.refractory.net](http://postgis.refractory.net). Data último acesso: 07 Junho, 2009.
- [18] Postgresql. <http://www.postgresql.org>. Data último acesso: 07 Junho, 2009.
- [19] Raster graphics. [http://en.wikipedia.org/wiki/Raster\\_Graphics](http://en.wikipedia.org/wiki/Raster_Graphics). Data último acesso: 13 Outubro, 2008.
- [20] Scalable vector graphics. <http://www.w3.org/Graphics/SVG>. Data último acesso: 30 Outubro, 2008.
- [21] Schema sld. <http://schemas.opengis.net/sld/>. Data último acesso: 12 Maio, 2009.
- [22] Shapefile. <http://en.wikipedia.org/wiki/Shapefile>. Data último acesso: 13 Outubro, 2008.
- [23] Sistema coordenadas geográficas. [http://pt.wikipedia.org/wiki/Coordenadas\\_geograficas](http://pt.wikipedia.org/wiki/Coordenadas_geograficas). Data último acesso: 11 Maio, 2009.
- [24] Sistemas informação. <http://pt.wikipedia.org/wiki/SistemasInformacao>. Data último acesso: 21 Maio, 2009.
- [25] Sistemas informação geográfica. [http://pt.wikipedia.org/wiki/Sistema\\_de\\_informacao\\_geografica](http://pt.wikipedia.org/wiki/Sistema_de_informacao_geografica). Data último acesso: 22 Maio, 2009.



- [26] Sistemas informação geográfica (tobler). [http://en.wikipedia.org/wiki/Waldo\\_R.\\_Tobler](http://en.wikipedia.org/wiki/Waldo_R._Tobler). Data último acesso: 22 Maio, 2009.
- [27] Styled layer descriptor. <http://www.opengeospatial.org/standards/sld>. Data último acesso: 3 Maio, 2009.
- [28] *Simple Feature Specification*. <http://www.opengeospatial.org/standards/sfs>. Data último acesso: 06 Junho, 2009.
- [29] Tutoriais web services. <http://www.netbeans.org/kb/docs/websvc/pet-catalog-screencast.html>. Data último acesso: 07 junho, 2009.
- [30] Vector graphics. [http://en.wikipedia.org/wiki/Vector\\_Graphics](http://en.wikipedia.org/wiki/Vector_Graphics). Data último acesso: 13 Outubro, 2008.
- [31] Web feature service. <http://www.opengeospatial.org/standards/wfs>. Data último acesso: 01 Maio, 2009.
- [32] Web map service. <http://www.opengeospatial.org/standards/wms>. Data último acesso: 2 Maio, 2009.
- [33] Xbase. <http://pt.wikipedia.org/wiki/XBase>. Data último acesso: 20 Maio, 2009.
- [34] Xhtml. <http://en.wikipedia.org/wiki/XHTML>. Data último acesso: 4 Novembro, 2008.
- [35] Xml. <http://en.wikipedia.org/wiki/XML>. Data último acesso: 4 Novembro, 2008.
- [36] Xslt. <http://www.w3.org/TR/xslt>. Data último acesso: 4 Novembro, 2008.
- [37] Ana.Vaz. Pesquisas interactivas para informação geo-referenciada em dispositivos móveis. Master's thesis, Faculdade de Ciências da Universidade de Lisboa, Portugal, 2008.
- [38] Andréas M. Winter A.Neumann. Time for svg - towards high quality interactive web-maps. Technical report, Institute of Cartography, ETH Zurich ETH Hoenggerberg, CH-8093 Zurich Freytag and Berndt, Austria Brunnerstr. 69, A-1231 Vienna, 2001. 20th Internationals Cartographic Congress, Beijing, 2001 - carto.net.

- [39] F.-J. Behr, L. Hui, C. Hang, F. Weldeleslasie, and C. Zhuoer. Phpmymwms: an open-source-based svg-oriented framework for extended web map services. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 6421 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, 2006.
- [40] P.A. Burrough and R.A McDonnell. Principles of geographical information systems. Oxford University , UK, United Kingdom, 1998.
- [41] Maria Beatriz Carmo, Ana Paula Afonso, Paulo P. de Matos, and Ana Vaz. Movisys - a visualization system for mobile devices. 2008. [http://xldb.di.fc.ul.pt/wiki/Ana\\_Paula\\_Afonso](http://xldb.di.fc.ul.pt/wiki/Ana_Paula_Afonso).
- [42] Maria Beatriz Carmo, Ana Paula Afonso, and Paulo Pombinho Matos. Visualization of geographic query results for small screen devices. In *GIR '07: Proceedings of the 4th ACM workshop on Geographical information retrieval*, pages 63–64, New York, NY, USA, 2007. ACM.
- [43] LI Hui. Design and implementation of a cartographic client application for mobile devices using svg tiny and j2me. Master's thesis, Univ. of Applied Sciences Stuttgart, 2006.
- [44] L.Chittaro. Visualizing information on mobile devices. *Computer*, 39(3):40–45, 2006.
- [45] A. Neumann. Use of svg and ecma script technology for e learning purposes. Technical report, Institute of Cartography, ETH Zurich, CH-8093 Zurich, Switzerland, 2005. ISPRS Workshop Commissions, Tools and Techniques for E Learning, Potsdam, 2005.
- [46] P.Pombinho. Visualização de informação geo-referenciada em dispositivos móveis. Master's thesis, Faculdade de Ciências da Universidade de Lisboa, Portugal, 2008.
- [47] S.Burigat and L.Chittaro. Visualizing the results of interactive queries for geographic data on mobile devices. pages 277–284, 2005.
- [48] T.Reichenbacher. Svg for adaptive visualisations in mobile situations. In *SVG open*, In Proc. Conference on *Scalable Vector Graphics (SVG Open)*, 2002.

